2000

# Bottom-up design of artificial neural network for single-lead electrocardiogram beat and rhythm classification

Srikanth Thiagarajan
*Louisiana Tech University*

# INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

Bell & Howell Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA

UMI®
800-521-0600

.

# NOTE TO USERS

**This reproduction is the best copy available**

**UMI**

# BOTTOM-UP DESIGN OF ARTIFICIAL NEURAL NETWORK FOR SINGLE LEAD ELECTROCARDIOGRAM BEAT AND RHYTHM CLASSIFICATION

by

Srikanth Thiagarajan, MS

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

COLLEGE OF ENGINEERING AND SCIENCE
LOUISIANA TECH UNIVERSITY

May 2000

LOUISIANA TECH UNIVERSITY

# THE GRADUATE SCHOOL

May 15, 2000
Date

We hereby recommend that the thesis prepared under our supervision by Srikanth Thiagarajan entitled Bottom-up Design of Artificial Neural Network for Single Lead Electrocardiogram Beat and Rhythm Classification be accepted in partial fulfillment of the requirements for the Degree of Doctor of Philosophy

Stan Napper
Supervisor of Thesis Research

Stan Napper
Head of Department

Biomedical Engineering
Department

Recommendation concurred in:

Steven A Jones

Paul Hamm

Advisory Committee

Approved:

Director of Graduate Studies

Approved:

Dean of the Graduate School

Dean of the College

GS Form 13
(1/00)

# ABSTRACT

Performance improvement in computerized Electrocardiogram (ECG) classification is vital to improve reliability in this life-saving technology. The non-linearly overlapping nature of the ECG classification task prevents the statistical and the syntactic procedures from reaching the maximum performance. A new approach, a neural network-based classification scheme, has been implemented in clinical ECG problems with much success. The focus, however, has been on narrow clinical problem domains and the implementations lacked engineering precision. An optimal utilization of frequency information was missing. This dissertation attempts to improve the accuracy of neural network-based single-lead (lead-II) ECG beat and rhythm classification. A bottom-up approach defined in terms of perfecting individual sub-systems to improve the over all system performance is used. Sub-systems include pre-processing, QRS detection and fiducial point estimations, feature calculations, and pattern classification. Inaccuracies in time-domain fiducial point estimations are overcome with the derivation of features in the frequency domain. Feature extraction in frequency domain is based on a spectral estimation technique (combination of simulation and subtraction of a normal beat). Auto-regressive spectral estimation methods yield a highly sensitive spectrum, providing several local features with information on beat classes like flutter, fibrillation, and noise. A total of 27 features, including 16 in time domain and 11 in frequency domain are calculated. The entire data

and problem are divided into four major groups, each group with inter-related beat classes. Classification of each group into related sub-classes is performed using smaller feed-forward neural networks. Input feature sub-set and the structure of each network are optimized using an iterative process. Optimal implementations of feed-forward neural networks provide high accuracy in beat classification. Associated neural networks are used for the more deterministic rhythm-classification task. An accuracy of more than 85% is achieved for all 13 classes included in this study. The system shows a graceful degradation in performance with increasing noise, as a result of the noise consideration in the design of every sub-system. Results indicate a neural network-based bottom-up design of single-lead ECG classification is able to provide very high accuracy, even in the presence of noise, flutter, and fibrillation.

# TABLE OF CONTENTS

# LIST OF TABLES

# LIST OF FIGURES

# NOMENCLATURE

| | |
|---|---|
| AFIB | Atrial fibrillation |
| AFL | Atrial flutter |
| ANN | Artificial Neural Networks |
| APC | Atrial premature contraction |
| AR | Auto-regressive |
| ARMA | Auto-regressive moving average |
| BIH | Beth-Israel Hospital |
| BP | Back-propagation |
| CSE | Common standards in quantitative electrocardiography |
| ECG | Electrocardiogram |
| FFT | Fast Fourier transform |
| FP | Fiducial point |
| LBBB | Left bundle branch block |
| MA | Moving average |
| MIT | Massachusetts Institute of Technology |
| MWI | Moving window integrator |
| NSR | Normal sinus rhythm |
| PR | Pattern recognition |
| PVC | Premature ventricular contraction |
| RBBB | Right bundle branch block |
| VBI | Ventricular bigeminy |
| VCOUP | Ventricular couplet |
| VTRI | Ventricular trigeminy |
| WPW | Wolff-Parkinson-White |
| WT | Wavelet transform |

# ACKNOWLEDGEMENTS

I would like to express my sincere thanks and gratitude to all those who have associated themselves with my research studies, aided me in my investigations, offered their valuable suggestions and made this dissertation possible.

I would like to express my sincere thanks and gratitude to:

Dr. S. A. Napper, my advisor for the continuous guidance and helping me maintain my focus.

My doctoral committee members, Dr. S.A. Jones, Dr. R.W. Schubert, Dr. H. Gu and Dr. P.N. Hale for their constant support and motivation.

Dr. C. J. Robinson and Dr. Puckett. F., for their enormous administrative support through Center for Biomedical Engineering and Rehabilitation Sciences and Dr. Nassar.R., and Dr. Zou for their technical input.

Cardiologists Dr. Misra. K.P., Dr. D. Prabhakar D.M. and Dr. Woodard and graduate students Dr. Ai Hua and Dr. Sridevi for their critical annotation of ECGs.

Administrative staff Celia.W., Sherry.J. and Sellers. B., for many timely help.

My lab mates Kim, Zhou chen, Venkatesh, Ramanathan, and Nagesh for creating an excellent atmosphere for research.

All of my friends and my well wishers and my family members for the encouragement and support offered to enable me to accomplish this objective.

I finally thank almighty god for providing me strength and spirit through out the work.

# CHAPTER 1

# INTRODUCTION

## 1.1. Computerized ECG Interpretation -- Overview

Computerized analysis of Electrocardiogram (ECG) signals has become common over the last decade. Clinicians and cardiologists accept computerized ECG interpretation as a much-needed supplement, and efforts at standardization have been taking place for more than two decades [Willems *et al.*, 1990]. Three main processes in any automated ECG interpretation system involve (i) ECG signal pre-processing, (ii) feature extraction, and (iii) pattern classification. The first step of signal pre-processing has been a relatively standard process, subjected to the requirements of the system. Newer algorithms have been developed for feature extraction from ECGs as well as for pattern recognition of ECG. In many programs described in research literature and from the scarce information disclosed by equipment manufacturers, the implementation of steps (ii) and (iii) seem to be independent.

Early developments in pattern classification were dominated by two main approaches: statistical and syntactic approaches. One of the earliest and the most popular ECG interpretation systems developed by Haisty *et al.*, 1972, made use of extensive feature calculations and statistical pattern-recognition methodology. Pipberger, who pioneered the development of computerized ECG interpretation systems, considered his system (Haisty *et al.*, 1972) as belonging to the second

generation of computerized ECG interpretation systems, whereas the first generation was dominated by the syntactic approach. Later, many of the commercial ECG interpretations systems followed variations of syntactic pattern recognition methodologies [Willems.J.J., 1990].

In spite of the improvements in the first two generations of ECG signal classification and interpretation, these systems lacked specificity and sensitivity, and all of the algorithms are machine specific. Yet, their performances continued to improve, and the main problems arose because of the inherent non-linearly separable nature of the ECG classification problem, a fact highlighted by MacFarlane and Lawrie, as early as in 1974 and more recently by Elghazzawi and Geheb, 1997, and Rautaharju *et al.* in 1992. An illustration of different approaches to ECG classification problem and their inter-relations is provided in Fig.1.1.

In the last two decades, a third methodology of pattern recognition, namely artificial neural networks (ANN), has become quite popular, and several successful implementations with respect to ECG pattern recognition have been reported [Baxt, 1992; Silipo *et al.*, 1995]. Unique applications such as detection of lead reversals in ECG recordings have been tried successfully using ANNs [Heden, 1996]. Heden *et al.*, 1997, have demonstrated that the performances of ANNs in predicting heart attack to be at least 10% superior to an individual expert.

Implementations of ANN-based ECG classification schemes have been restricted to problems of narrow clinical domain, and the focus has been on applications rather than on design concepts or choice of better features. With newer knowledge available from the theory of ANNs and with the variety of accurate features, the stage is

now set for moving towards a bottom-up approach in designing artificial neural network-based clinical ECG interpretation systems. A bottom-up approach is defined in terms of need-based implementation of component sub-systems; *i.e.*, QRS complex detection, fiducial point estimation, feature extraction, beat classification, and rhythm identification. The bottom-up approach has already been suggested for syntactic approach by many authors, which has led to dramatic performance increase in such systems [Steinberg *et al.*, 1962; Trahanias and Skordalakis, 1989].

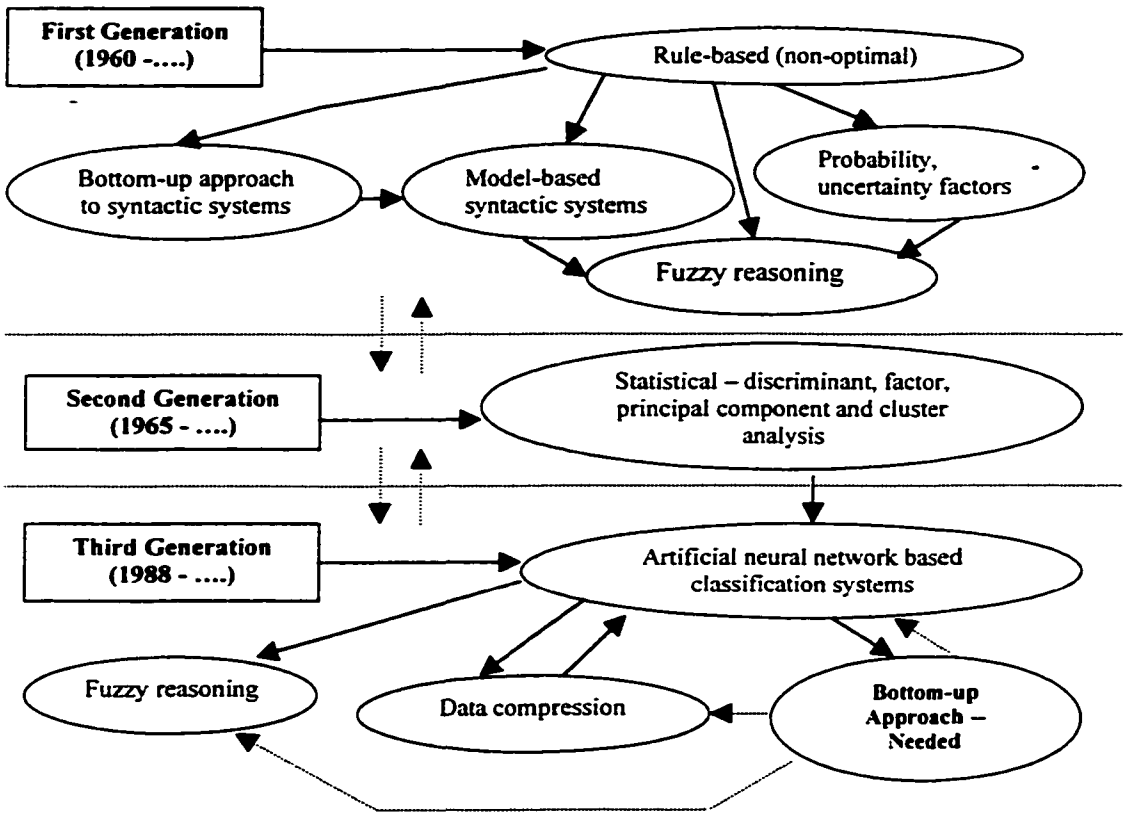



Fig.1.1. Evolution of ECG classification systems. – broken lines indicate indirect evolution and the connected lines indicate direct evolution

The requirements of similar ANN-based ECG classification systems include

(i) The ability to accommodate morphological variations in the ECG beats including noisy beats, and

(ii) An adaptive methodology to standardize and extract the features available from the ECG acquired by instrumentation with varying specifications.

The proposed research deals with such implementation strategies, following a bottom-up solution for the computerized ECG interpretation problem.

## 1.2. Hypothesis and Objective

The hypothesis of the proposed dissertation is that a bottom-up approach in designing an ANN-based ECG interpretation system will improve the system performance and result in an optimized and expandable design.

The objective of the present work is to present an integrated methodology of single-lead ECG beat classification, for any short-term routine clinical data. The bottom-up approach focuses on the optimization of individual sub-systems, resulting in the overall improvement in system performance. The strategy is to combine feature extraction and pattern classification steps into one optimized module, with the final design expandable for the more exhaustive problem of designing multi-lead ECG interpretation systems. Optimal structures for artificial neural networks with feed forward connections and associative networks will be the pattern classifiers. Optimality in neural network implementations is defined with respect to choice of minimal set of input features, and the least number of hidden neurons.

Routine ECG data from lead-II electrode position acquired from different environments are used for the study. Data segments of six seconds to ten seconds duration used in the study are annotated for all beats by two independent cardiologists.

## 1.3. Research Need

Algorithms developed for commercial systems, the most successful, can be classified into two generations, as done by Haisty *et al.*, [1975]. The first generation of computerized ECG interpretation systems is basically syntactic pattern classifiers. A classic example of first generation systems is the one developed by Pipberger himself along with Klingeman [1961].

Later, such systems evolved into expert systems, with separate knowledge bases and inference engines. The knowledge base essentially consists of "If-Then" statements, with specific thresholds for parameter values, not all of them scientifically determined. Degani and Bortolan [1990], question the use of such binary logic. To overcome the drawbacks of applying deterministic solutions to an essentially non-deterministic problem, later algorithms used certainty factors and other probabilistic measures. Degani and Bortolan [1990] followed the other option of fuzzy logic. However, all such approaches end up being deterministic since the algorithms do not evolve with the data. Such algorithms present problems like inconsistent performance for similar data and inability to adapt to data recorded under slightly different conditions.

Elghazzawi and Geheb [1997], point out another major problem associated with heuristic rule-based approaches in an important critique. They found that the problem of distinguishing between normal (N) and ventricular (V) beats belongs to the non-linearly separable class. When AND-OR binary structures with hand-tuned thresholds or linear-separation techniques are used to separate N and V distributions in a K-dimensional feature space, errors are guaranteed because these distributions are not

linearly separable. An overlap exists between the ranges of feature values for the two classes. As a result, these algorithms have a limited dynamic range. The designers of these techniques always faced the dilemma of tuning the algorithms to be more sensitive at the expense of achieving less positive predictivity, resulting in a high rate of ventricular false-positives and vice versa. Such problems exist through out the ECG interpretation task [Rautaharju *et al.*, 1992].

Compared to the disadvantages of first-generation programs, second-generation programs based on statistical methodologies present a different set of problems. They start with the assumption of Gaussian characteristics for the signal as well as the features derived from ECG signal. Discrete ECG features may not always obey Gaussian distribution as suggested in a review of bio-signal interpretation by Ciaccio *et al.* [1993].

Another problem in statistical pattern classification relates to the complexity of the problem domain. With multiple output classes and input features, linear discriminant rules begin to fail. Non-linear statistics is not yet fully understood, and this fact is especially evident in a clinical environment. These causes lead to the unpopularity of the second-generation programs among the end-users, cardiologists and clinicians. Another approach involved model-based arrhythmia analysis, as defined by Tong [1992]. EINTHOVEN used the information from single-lead ECG data and created hypotheses about the rhythm classification. In spite of providing improvement over pure knowledge-based, syntactic methods, EINTHOVEN remains nevertheless a syntactic, rule-based approach. Consequently, the inability to separate non-linearly separable cases remains. Further, EINTHOVEN starts from the models, making use of

descriptive features, omitting the more fundamental problem of feature extraction. Any modern automated ECG interpretation system should take into account the problems facing feature extraction and try to estimate the necessary features from the raw data, including a few noisy beats.

Although approximately 30-40 different computer programs have been developed for ECG beat classification or for related electrophysiology interpretation, many of these systems are proprietary and are developed by industry. Most of the algorithms deal with arrhythmia identification in multi-lead settings. Evaluation of such programs is not available for outside researchers. Such conditions have resulted in continuing apathy towards computerized interpretation by physicians, though the improvement in the quality of the signal due to digital technology is much appreciated [Leblanc, 1989]. These factors call for design of systems by clinicians and physicians themselves. Design of artificial neural networks-based ECG interpretation systems is one typical example. Several applications have originated from cardiologists and clinicians using their own neural network based designs [Baxt, 1992; Devine and Macfarlane, 1993]. The design of such systems focuses only on the neural networks aspect, and ECG data along with features are derived from commercial machines. This approach, in turn, results in a non-optimal design for ECG interpretation.

An optimal design should have proper transfer of control between the feature extraction and the pattern classifier steps. Feature extraction includes the fiducial point detection and the estimation of time and frequency domain features. Pattern classifier performs the beat and rhythm classifications. Similarly, the feature extraction procedure itself can be designed with time and frequency domain features extracted

depending on the need. Pattern recognition can be done step-by-step by using the minimal number of classes and data sets, and improvising upon the initial design. Such a concentrated effort on minute details of implementation will result in a system that can be expanded to tackle multi-lead, disease-specific ECG interpretation problems. Addition of clinical details becomes the next logical evolution in this process. Beat identification should include morphological variations like ST slope changes, and paced beats, apart from arrhythmia conditions. Present research aims to provide the basic foundation to such a bottom-up approach by focusing on single lead ECG interpretation in relation to multiple morphological and rhythm variations.

## 1.4. Organization of the Thesis

Present dissertation is an attempt to provide an efficient bottom-up design for the classification of single lead ECG beat and rhythm classifications. Artificial neural networks are the pattern classifiers in the present dissertation. The present dissertation is divided into four more chapters focussing on background, design and methodology, results and conclusions respectively. Chapter 2 provides a brief overview on vital topics and techniques related to ECG signal processing and classification. The chapter begins with a brief overview on the clinical role of ECG. The chapter also summarizes vital developments in ECG signal processing and computerized ECG interpretation systems over the last two decades. Signal processing aspects of the present work, especially those related to extracting features in both time and frequency domains are explained in chapter 3. The methodology of computerized ECG interpretation that derives concepts from several fields (e.g., electrocardiology, digital signal processing,

and pattern recognition) is organized into several smaller sections, finally arriving at a unified design.

Chapter 3 on design and methodology and chapter 4 on results are arranged into sections, which directly correlate. Sections in design and methodology deal with sub-tasks in present dissertation work and the chapter on results presents corresponding results for those sub-tasks. The document ends with chapter 5 on conclusions derived from the present research. Chapter 5 also provides a model for expanding the present work into a typical multi-lead ECG interpretation problem. Appendices are included to provide the reader with details of the implementation.

# CHAPTER 2

# BACKGROUND

## 2.1. Clinical Role of Electrocardiogram - Brief Overview

The electrocardiogram (ECG) has been the major diagnostic tool for cardiologists and the ECG signal provides almost all information about the electrical activity of the heart. With nearly 40% of the deaths in the US in 1996 occurring due to cardiovascular diseases (CVD), and with around 18% of the population having one form of CVD or the other in US, cheaper and earlier diagnosis of symptoms is needed [AHA annual report, 1999]. The scene is not much different globally. Simple routine ECG analysis has always been the indicator of first clues in emergency medicine, in order to provide the treatment needed in case of the problems related to heart [Sheffield, 1987]. Arguably, the electrocardiogram (ECG) signal is one of the most studied waveforms.

The ECG signal provides a recording of body surface potentials generated by the heart and is obtained from electrodes placed on the surface of the body. An ECG is thus a plot of the time-dependence of changing potential differences between electrodes on the body surface. The electrical activation precedes the mechanical pumping action of the heart. The 12-lead ECG waveforms pertaining to the normal heart are fairly standardized and are shown in Fig. 2.1., and a diagram of a single lead ECG (lead II)

with vital wave components is shown in Fig. 2.2. Table.2.1. provides information on each lead position.

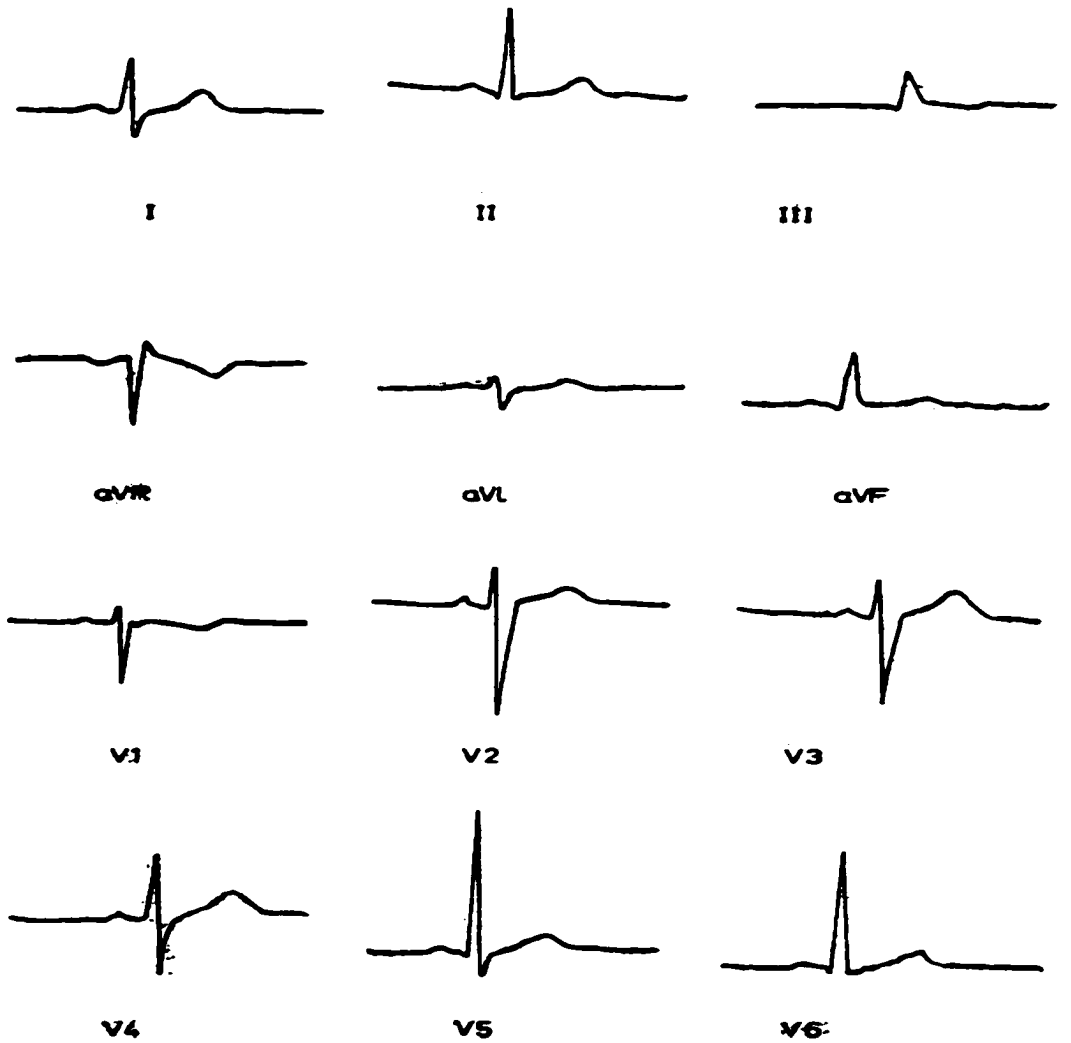



Fig. 2.1. Standard 12-lead ECG for a normal heart.[MacFarlane and Lawrie, 1974]

The features usually observed by a physician or a cardiologist, are shown in Fig. 2.2. These features are called local features and are the ones easily observable for a trained physician [Kelly, 1984]. Each component in an ECG beat signifies the excitation and the spread of electrical activity inside the heart. Table. 2.1, provides the relation between the observed segment in ECG and the corresponding electrical activity

of a normal heart [Katz. A.M., 1992]. Table.2.2. provides the information about the different leads in a 12-lead acquisition system.

Table.2.1. Electrical activities corresponding to the component waves

| Component | Activity |
|---|---|
| P wave | Atrial Depolarization |
| PR Interval | Atrioventricular Conduction |
| QRS Complex | Ventricular Depolarization |
| S-T Segment | Plateau of the Ventricular Action Potential |
| T wave | Ventricular Repolarization |

Table. 2.2. Information on 12-Lead system

| Leads | Polarity | Position |
|---|---|---|
| Lead I | Bipolar lead | left arm (+) & right arm (-) |
| Lead II | Bipolar lead | Left leg (+) & right arm (-) |
| Lead III | Bipolar lead | Left leg (+) & left arm (-) |
| Lead aVL | Unipolar | Left arm (+) |
| Lead aVR | Unipolar | Right arm (+) |
| Lead aVF | Unipolar | Left leg (+) |
| Lead V1 | Unipolar | Right side of sternum (+) |
| Lead V2 | Unipolar | Left side of sternum (+) |
| Lead V3 | Unipolar | Between V2 and V4 (+) |
| Lead V4 | Unipolar | Mid-clavicular line (+) |
| Lead V5 | Unipolar | Anterior line (+) |
| Lead V6 | Unipolar | Mid-axillary line (+) |

In clinical practice, a 12-lead ECG provides more information compared with a single lead ECG. This result is similar to taking pictures of an automobile from different angles. However, any new methodology for automating ECG classification starts usually on single lead data sets and the procedure extended to multi-lead systems, especially for the initial steps of signal pre-processing and feature extraction [Laguna, 1994; Skordalakis, 1986; Willems, 1990]. Though single-lead ECG interpretation by no means represents a complete ECG signal interpretation problem [Hatala *et al.*, 1999], standardization and optimization of this step is needed for expanding the solution to multi-lead ECG interpretation as well as for getting other clinical information. According to MacFarlane and Lawrie [1974], the use of only one lead for interpreting rhythms has not proved a major disadvantage especially for detecting most of the common rhythm disturbances.

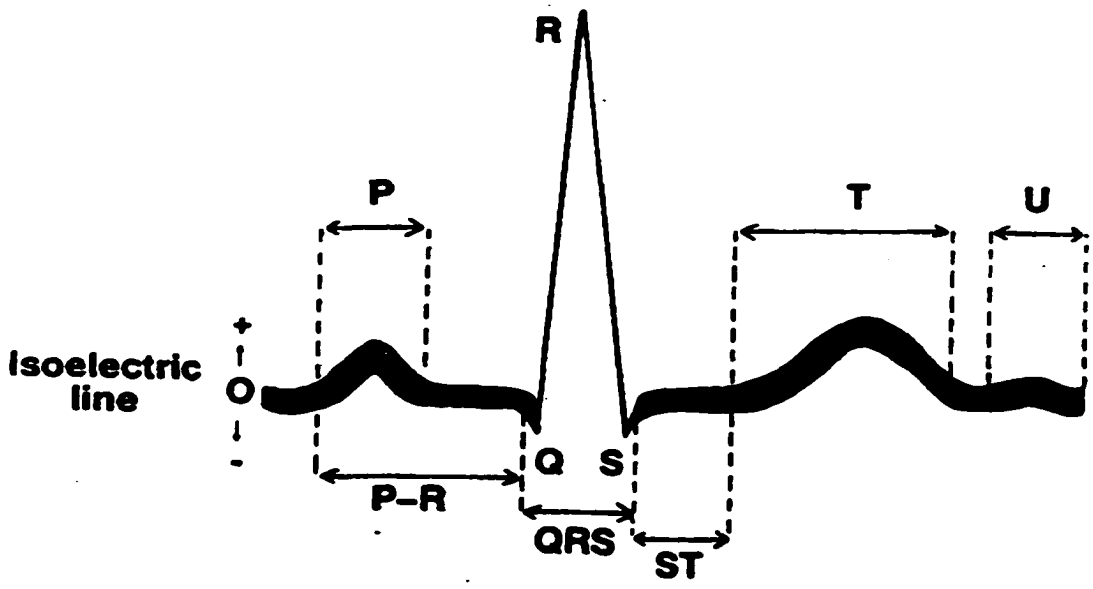



Fig. 2.2. Normal ECG beat with the vital intervals and amplitudes indicated. [Kelly, 1984]

Electrical conduction points and pathways in a heart are shown in Fig. 2.3. and the pathways are explained in Fig. 2.4. in the form of a schematic of a cardiac conduction model. The abnormal patterns seen in any ECG beat may belong to one of several arrhythmia classes or morphological variations, or may be just electrical noise. An arrhythmia results when the sinus rhythm fails, is disturbed, or is usurped during its quiescent period, or its conduction is interfered with. Two broad categories of arrhythmia have been described [Conover, 1992]:

(1) Abnormalities of conduction caused by conduction block, reentry, or reflection, and

(2) Abnormalities of impulse initiation caused by altered automaticity or triggered activity.

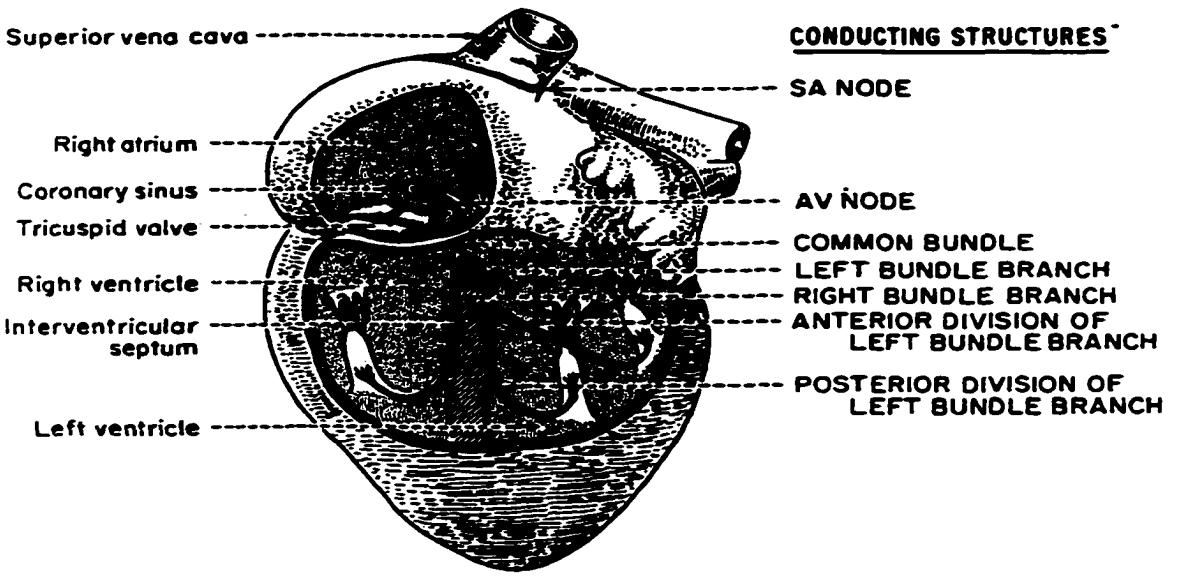



Fig. 2.3. Electrical conduction points and pathways in a heart [Katz, 1992].

Many rhythm and morphological variations are easy to identify, and they manifest in the form of differences in time intervals, amplitudes, and shapes of QRS complex, and P and T waves. Sometimes, one of the signal components is absent, as in an absent P wave in premature ventricular contraction. Further, several wave

abnormalities cannot be attributed to any single parameter, but rather a combination of them. For example, in the simple case of sinus tachycardia, heart rate, PR interval, QT interval and P wave have all provided classical information about the arrhythmia [Wagner. G.S., 1994].

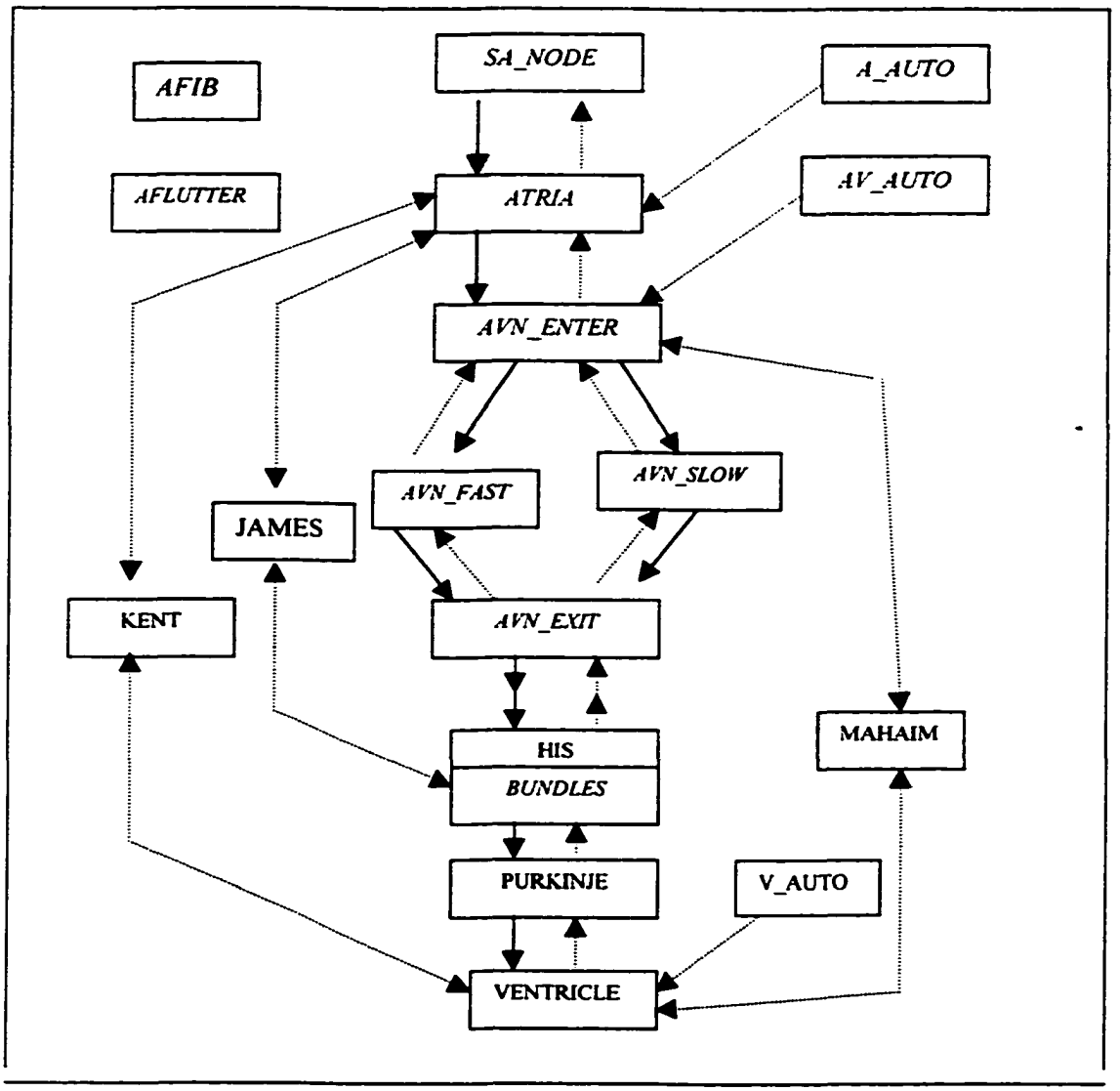



Fig. 2.4. Schematic representation of the cardiac conduction model. Solid lines, normal conductive pathways; dashed lines, abnormal conductive pathways.

As suggested by Garfein [1990], the electrocardiograph is capable of supplying much more information than has been extracted from it in the past. The quantitative

and qualitative assessment of the activity of the autonomic nervous system and indirect assessment of ventricular function are all possible in the power spectrum analysis of heart beat variation. Definitely, routine electrocardiogram recordings are here to stay and their clinical significance is increasing in spite of the innovative imaging and other diagnostic modalities. Digitization of electrocardiogram signals and its subsequent processing has created a clearer ECG signal, with much more accurate characteristics [Jain and Rautaharju, 1980]. The accuracy of features has clearly increased, and it is natural to consider the automation of the process.

## 2.2. Computerized ECG Pattern Recognition

Although the first attempts to automate electrocardiogram analysis by digital computer were made as early as 1961 by H.V. Pipberger and coworkers, it took considerably more time to develop operational computer programs than originally anticipated. However, over the last two decades, computer processing of clinical ECGs has increased rapidly. Until recently there were no standards in this field. There were no common definitions of waves, no standards of measurement or diagnostic classification, and no uniform terminology for reporting. Due to the lack of standards, data processed by one machine could not be transmitted and read by another machine [Willems *et al.*, 1990]. Efforts are going on to create such standards in data formats and features, as well as in the performance of the algorithms. Single-lead databases are already available commercially and a massive effort involving multiple clinics is going on for the standards in 12-lead systems [Rautaharju *et al.*, 1992].

Three types of statements are possible from computerized ECG interpretation systems, as suggested by the American Heart Association as well as in the Common

Standards in Quantitative Electrocardiography study [Willems, 1990; Jain and Rautaharju, 1980]. Type-A statements refer to pathophysiological states or anatomic lesions, *e.g.*, detection of myocardial infarct using 12-lead ECGs [Baxt, 1992]. Their accuracy can be verified by ECG-independent information. Type-B statements refer to the electrophysiological conditions from the ECG. A sample Type-B statement may look like: Wolff-Parkinson-White Syndrome (WPW) with possible right anteroseptal pathway. Type-C statements are about ECG features not necessarily correlated with physiological (or) pathological states, as in the present study. Example of a Type C statement: ECG recording shows unifocal, ventricular bigeminy [Svirbely, and Sriram, 2000]. Type-A databases require multi-clinic studies for neural network implementations, due to the requirements of massive databases of subjects with disease conditions.

Illustration of the relation between the three types of statements from the ECG machines is provided in Fig. 2.5. For example, relating single lead of information to a multiple lead configuration involves adding another dimension to the original information, some of the information might be redundant. Similarly, the relation between the information available from multi-lead ECG data (Type-B) and the combined information available from the multi-lead ECG data and clinical data (Type-A) may also be redundant.

Borovsky and Zywietz, [1980], formalized the relations between feature extraction and pattern classification for ECG signal interpretation problems, as shown in Fig. 2.6. Feature extraction is a process of reducing the dimensionality of the feature space in a problem. Usually, the dimension of the feature vector Y is less than the

original pattern vector X (M<N). The process may be as simple as standard fiducial point selection or may be some transformation T[X]. The transformations can be linear or non-linear. The next step, feature selection, is optional and means further data reduction through extraction of parameters for comparison and classification. The selected features Z may be a subset of the feature vector Y and are fed into the pattern classifier. Some features might be discarded due to redundancy with other features.

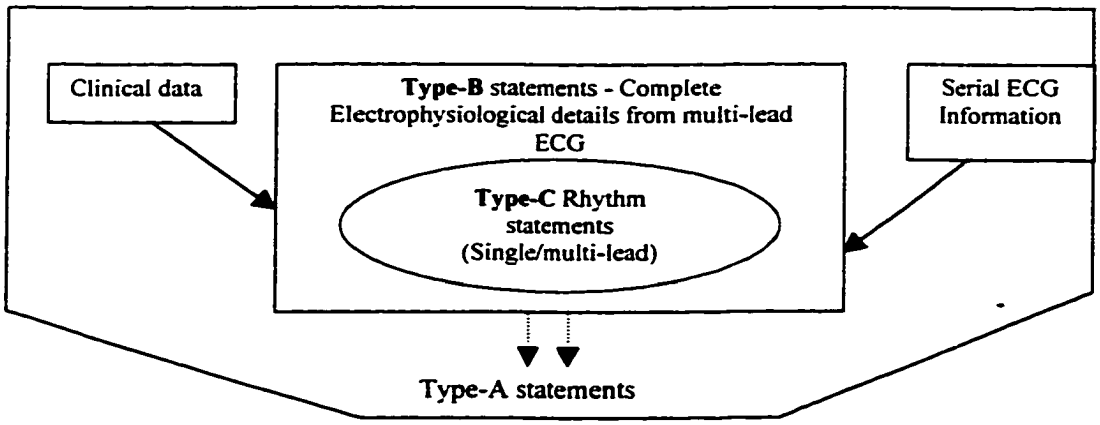



Fig. 2.5. Relation between the Type-A, Type-B and Type-C systems with respect to classification task

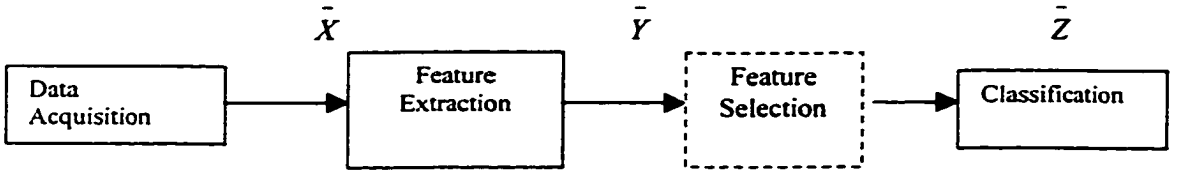



Fig. 2.6. The relations between feature extraction and pattern classification for ECG signal interpretation

Though the overall block structure of the 12-lead ECG interpretation systems have not changed much in the last four decades, internal implementations have shown significant developments. Pipberger *et al.*, [1961], used syntactic rules to interpret 12-lead ECG and clinical data to detect infarction. A large group of ECG interpretation systems followed suit [Leblanc, 1989; Belforte *et al.*, 1979]. The first generation of rule-based ECG interpretation systems was soon followed by statistical pattern analysis

methodologies, which can be considered as second-generation systems [Haisty *et al.*, 1972]. Statistical methods used both clustering algorithms as well as discriminant analysis algorithms. Pipberger, in his pioneering research, used a large number of features and applied a set of principles, like feature extraction, and feature selection and elimination and classification, which have been followed ever since. However, limitations arose due to inherent inaccuracies in features and unavailability of frequency domain analysis techniques [MacFarlane and Lawrie, 1974].

Meanwhile, the development process was taken over by commercial domain, and for nearly 15 years, results of the performance evaluations were not published. Efforts in industry focused on shifting the software into microprocessor based systems and on reducing the size of ECG machines [Medtronic 9790 programmer; Marquette ECG Machines]. During this period, at least 25 independent groups of researchers were active in perfecting the overall technology of computerized ECG interpretation and a need was felt for setting standards in the U.S.A., Europe, and Japan. In the early 1980's, a committee on Common Standards in Quantitative Electrocardiology (CSE) was set up in Europe in collaboration with American Heart Association, and some institutes in Japan and the U.S.A [Willems, 1990; Okajima *et al.*, 1990; Degani and Bortolan, 1990].

A standardized database was the first need for any proper comparison in such a huge statistical task [Laguna *et al.*, 1994; Moody, 1992] and parallel efforts took place at MIT-Harvard University, Boston and at many European member clinics, starting in late 1970's. These massive efforts resulted in standardized databases, which in turn provided a scientific approach for testing the signal processing and pattern classification algorithms.

Quite a few databases have resulted from above efforts. The earliest one is the Massachusetts Institute of Technology-Beth Israel Hospital (MIT-BIH) database, which contains a set of two lead recordings with limb lead II and usually $V_2$. All the beats are annotated, and this database is considered the standard one for evaluation of algorithms for detecting common arrhythmia. MIT-BIH ECG database is a Type-C database, because it can only used to verify the algorithms for detecting various rhythms. Another database, much more comprehensive, albeit expensive, is the Common Standards for Quantitative Electrocardiography (CSE) diagnostic database. This database provides a combination of 3-lead as well as 12-lead information and belongs to Type-A & Type-B databases, having clinical details along with ECG data. Such databases are needed for comprehensive testing of overall multi-lead ECG interpretation systems.

Leblanc, [1989], in his review on quantitative arrhythmia, reported about the lack of standard criteria for evaluation and dependence on the instrumentation. Initial IBM developments focused on short recordings of 5-10s duration and focused on mainly P and QRS locations. Most of the initial algorithms like IBM, American Veterans Administration (AVA) *etc.*, had too many arbitrary threshold values [Degani and Bortolon, 1990]. Besides IBM and American Veterans Administrations (AVA) algorithms, other algorithms which took part in CSE evaluation are Nagoya, NOVACODE, AZTEC, Marquette, Louvain, Hannover, HP, IBM, Glasgow, Padova, Modular and Leuven programs [Willems *et al.*, 1990; Rautaharju *et al.*, 1998]. All these algorithms dealt with either 3-lead or 12-lead ECG. The aims of the CSE diagnostic study are to evaluate and eventually improve diagnostic ECG systems, to

compare different diagnostic strategies, and to establish an ECG-independent and clinically validated library to carry out these objectives. Such a massive study was only possible in a multi-center setting.

The history of computerized ECG interpretation may not be complete without a brief look at some of the algorithms that were evaluated by CSE group. Yanowitz *et al.*, [1974], presented their criteria applied on a single ECG lead sampled at 500 samples per second and preprocessed by the well-known AZTEC algorithms. Most of the decisions were based on RR interval and QRS duration assessments. For example, "normal" QRS duration was 90 milliseconds (user-adjustable), and prematurity was 12.5% or more with respect to a running average. A wide QRS (30 milliseconds wider than normal duration), which is premature, is a premature ventricular beat (PVB). Three or more PVBs in a row, at a rate of 100 beats per minute were defined as ventricular tachycardia (VT).

Belforte *et al.*, [1979], presented a syntactic method for analyzing ECG waveforms in a multi-channel setting. These authors believed that the design took advantage of the redundancy implicit in multichannel information systems. According to the researchers, human inspection of ECG waveforms is primarily an extraction of structural and qualitative information. The thesis author agrees with the above observation, but the way humans interpret the extracted information is assumed more parallel than syntactic. The authors themselves admitted the need to take care of ambiguities (non-linearly separable) by introducing fuzzy or stochastic linguistic variables processed by fuzzy or stochastic automata.

A decision-tree approach, a variant of "if-then" rule based methodology, is also followed in the Nagoya program, the most popular algorithm in Japan [Okajima *et al.*, 1990]. A realistic outcome came only after comparisons with other algorithms in CSE study. The Nagoya program dealt with a multi-lead ECG data duration of 9.6 seconds. Syntactic algorithms, despite their good performance, are all custom-oriented instead of establishing a generalized approach aimed at clarifying and standardizing the definition of cardiac arrhythmia. Arrhythmia decision logic in such systems is fairly simple, and most of the efforts have been devoted to the signal-processing logic [Leblanc, 1989].

Rautaharju *et al.*, [1992], discuss the serial ECG classification system for clinical trials and epidemiological studies. This work considers the evolution of the feature values over time. The main advantage of the NOVACODE logic is that it extracts from each record in one step all waveform information needed for coding of serial changes and for verification of significance of these changes. In further developments, Rautaharju *et al.*, 1998, used a hierarchical classification code where the term ECG abnormalities mean both arrhythmia as well as ECG categories associated with myocardial ischemia/infarction, hypertrophy, and fascicular blocks. Evaluation using the CSE data showed that the performance of the above algorithms is very good in terms of sensitivity but seems to lack specificity, due to conservative thresholding. Nearly all algorithms showed better diagnostic capability compared with individual cardiologists [Willems, 1990].

In an important review on arrhythmia detectors based on heuristic rules, Elghazzawi and Geheb, [1997], found that the problem of discrimination between normal (N) and ventricular (V) beats belong to the non-linearly separable class. When

linear-separation techniques with hand-tuned thresholds are used to separate N and V distributions in a k-dimensional feature space, errors are guaranteed because these distributions are not linearly separable. As a result, these algorithms have a limited dynamic range. The designers of these techniques are always faced with the dilemma of tuning the algorithms to be more sensitive at the expense of achieving less positive predictivity, resulting in a high rate of ventricular false-positives and vice versa. Such factors along with problems like inconsistency, and lack of standards in methodology have resulted in a decrease in popularity of syntactic approaches. A simple example for a non-linearly separable problem is illustrated with a classical EX-OR problem and its variants as shown in Fig. 2.7.

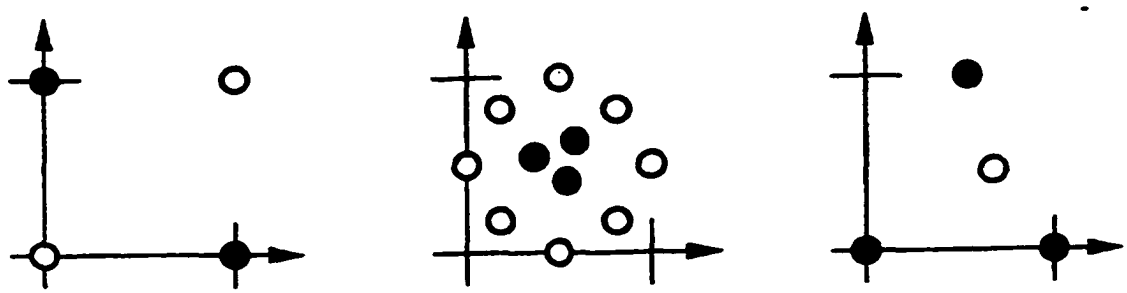


Fig. 2.7. EX-OR output classification complexity

Shaded and open circles indicate members belonging to two different classes represented with respect to two different feature axes. It is difficult to separate the two types of boxes with straight lines. Errors are unavoidable when using a single straight line to separate the classes. One solution for such problems is to use a model-based reasoning, which can make use of both the shallow as well as deep reasoning about the cardiac system [Tong, 1992]. However, taking care of all the uncertainties associated with a normal clinical recording is beyond such systems. In fact, only non-noisy

recordings are selected as inputs for EINTHOVEN, a model-based system designed by Tong. Further, model-based systems, being rule-based systems, derive essentially all their inherent limitations related to non-linearly overlapping classes. Another major drawback in EINTHOVEN is in relation to factors concerning the features. Descriptive features are used in the work, but no effort is taken to perfect the detection and estimation of features. Separating the feature extraction from the decision-making process affects the performance of the system. Signal pre-processing forms the starting point for the feature extraction.

## 2.3. Signal Pre-processing

Pre-processing of ECG signals for pattern recognition tasks involves band-pass filtering, detrending, removal of 50 or 60 Hz noise, and annotations of beats as well as rhythms. In the present work, pre-processing procedures need to take into account the signal characteristics of at least three data sets. Such multi-center, multi-machine data need to be standardized in terms of amplitude, analog-to-digital converter (ADC), as well as frequency characteristics. Sampling frequencies for different machines varies widely. Usual standards in sampling frequency in routine ECG monitors vary between 200 Hz to 1000 Hz. The most widely accepted value as of now is 500 Hz [van Bemmel *et al.*, 1990].

Another vital pre-processing task involves annotating the data set for training and testing. Annotated databases are needed for any classification task. Performance evaluation of pattern recognition algorithms needs comparison with the performance of manual expert annotations [Jenkins, 1998]. Laguna *et al.*, [1997], developed a database for evaluation of algorithms for measurement of QT and other waveform intervals in the

ECG. Manual annotations were made using a SUN workstation using WAVE, a unix based software tool. A similar idea using a PC and Matlab®-based interactive set-up is utilized in the present work. Comparison with physicians is the usual method for assessing Type-B statements, such as conduction disturbances and arrhythmia and for Type-C descriptive statements, such as axis or ST-T statements [Willems *et al.*, 1990].

Artificial recordings created by combining strings of identical selected beats as suggested in Common Standards for Quantitative Electrocardiography (CSE) report by Willems, [1990]. This process sometimes creates extra sample data for some rare conditions. Such artificial beats are used in many studies, in addition to actual beats. Simulation of ECG beats is sometimes suggested as an alternative to test the algorithms [Sommo *et al.*, 1980], in case the databases are unavailable.

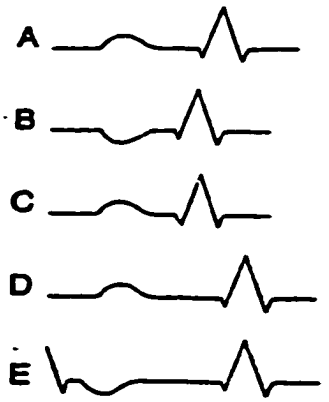
## 2.4. Developments in QRS Detection

QRS complex detection is vital in the context of both short and long term monitoring. Errors in QRS complex detection may become cumulative and may affect the entire feature extraction procedure. The QRS complex is usually the most prominent wave and R points are the prominent points in a beat and the easier indicators for calculating a beat interval.

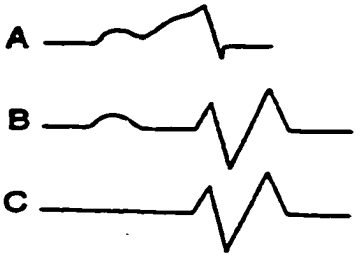
### 2.4.1. QRS Complex Morphology

A variety of QRS complex and T wave variations in morphology seen in lead II configuration are shown in Fig. 2.8., [Wagner, 1994]. The QRS complex is composed of higher frequency components than are the T and P waves, thereby causing the

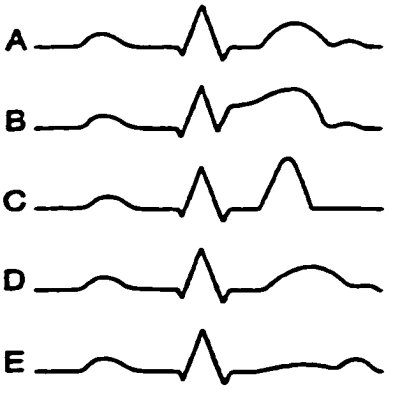
contour to be peaked rather than rounded. Positive and negative deflections of the QRS complex are assigned specific labels.




A. Normal P wave morphology, PR interval and QRS complex B.Impulse formed in AV node causing inverted P wave C. Normal P wave morphology and shorter PR interval D. Normal P wave and longer PR interval E. Abnormal P wave and longer PR interval, indicating retrograde conduction




A. Short PR interval and abnormal QRS complex B. Normal PR interval and abnormal QRS complex and C. Absent P waves




A. Normal ST segment B. Elevation of the ST segment C. Tall and peaked T wave D. Long $QT_c$ interval and E. Flat T wave and tall U wave [Wagner, 1994]

Fig. 2.8. Some of the morphological variations in QRS complex, ST segment and T waves in ECG beats for lead-II ECG [Wagner, 1994]

The normal Q wave is defined as the negative wave at the onset of the QRS complex. The R wave is the first positive wave appearing in a QRS complex; it may appear at the onset of the QRS complex or following a Q wave. The S wave is the negative wave following the R wave. Sharp rises and falls characterize the QR and RS segments. Actual data may include extraneous noise as well as other disturbances.

2.4.2. Detection Methodologies

Pahlm and Sornmo, 1984, made a review of software QRS detection in ambulatory monitoring. Many concepts in ambulatory monitoring also hold true for short-time clinical recordings. QRS detection, and the subsequent R point identification, becomes the start of estimation of other fiducial points, especially in routine arrhythmia monitoring algorithms. A general signal-processing approach of QRS detection involves a non-linear transformation and decision logic. Different non-linear transformations compared in the study included short-term energy calculation, calculation of smoothed envelope, parametric modeling of ECG signal, and maximizing the rectified, first difference between samples. Among the methods compared, performance of short-term energy calculations seemed to be better. Another interesting approach eliminates high T waves by using the first difference and maximizing the output.

Pan and Tompkins, [1985], formalized the short-term energy concept by their famous moving window integrator algorithm and standardizing the algorithm with MIT-BIH database. The algorithm with so many adaptive thresholds works well with ambulatory data. Absence of a large number of training beats hampers the implementation for routine short-term QRS detection.

Rasiah *et al.*, [1995], as well as Xue *et al.*, [1992], developed artificial neural network based QRS complex detection algorithms. Since the whole ECG signal is a non-linear signal generated from a non-linear system [Schimminger, 1998], Xue *et al.*, [1992], felt that ANN-based QRS detection algorithms would be the ideal choice. Keselbrener *et al.*, [1997] devised a simple and easily implantable method for R wave detection from ECG signals. The method is based on the subtraction of a filtered version of the signal. The filter used is a non-linear median filter. The median filter provides a smooth signal without R waves and after subtraction from the original signal, presents undistorted R-waves. Among new methods, wavelet transform-based methods provide good accuracy, though complexity of their implementations will likely create problems for clinicians [Li *et al.*, 1995].

## 2.5. Standards in Fiducial Point Calculation

T and P wave detection for each beat usually follow QRS complex detection. Direct measurement of the various intervals like the RR interval and the PR interval *etc.*, requires knowledge of the locations of the boundaries (the onsets and ends) of the P, QRS, and T waves. Broad steps in single-lead implementation are fibrillation process rejection, waveform location, and waveform boundary delineation [Laguna *et al.*, 1994].

### 2.5.1. Characteristics of P and T waves and other ECG Segments

At either slow or normal heart rates, the small, rounded P waves clearly occur before the taller, more peaked QRS complex. At more rapid rates, however, P wave may merge with the preceding T wave and become difficult to identify. The P wave

contour is entirely smooth, and it is usually monophasic, in all leads, except V1. The duration of P wave duration is normally less than 0.12 sec. Owing to its small size and lack of spectral variations for time varying characteristics of P wave, the detection of P wave remains a challenge. The PR interval is more commonly used than the P wave duration alone. The PR interval measures the time required for the impulse to travel from atrial myocardium adjacent to the SA node to the ventricular myocardium, and to the fibers of the purkinje network. A major portion of the PR interval reflects the slow conduction in the AV node, which is controlled by sympathetic-parasympathetic balance with in the autonomic nervous system.

The ST-segment represents the period of time when the ventricular myocardium remains in an activated or depolarized state. The junction between ST segment and QRS complex is called the J point and an angle of 90 degrees is formed between the two. The ST segment, then proceeds horizontally until it curves gently into the T wave. The length of the ST segment is influenced by factors that alter the duration of ventricular activation. Points along the ST segment are designated with reference to the number of milliseconds beyond the J point such as "J+20", "J+40", *etc.*

The first section of the ST segment is normally located at the same horizontal level as the baseline formed by PR segment discussed above and TP segment that fills in space between two beats. The appearance of the ST segment may also be altered during exercise or when there is an altered sequence of activation of the ventricular myocardium.

The smooth, rounded shape of a T wave resembles that of the P wave. But, there is greater normal variation of monophasic versus diphasic appearance. The initial

deflection of T waves is typically greater than the terminal deflection, producing a slightly "asymmetric shape," and "notching" of T waves is common in children [Wagner, 1994; Katz, 1992]. The U wave is normally either absent or present as a small, round wave following the T wave. It is normally in the same direction of the T wave, approximately 10% of its amplitude. Its mechanism is unknown, but it becomes taller in hypokalemia and inverted in heart diseases. Many of the rhythm variations as seen in actual data are shown in Appendix A, along with their definitions.

2.5.2. Strategies in Estimation of Fiducial Points

The detection of Q wave, S and J points usually involve detection of change in sign of the first difference, angle calculations or directional calculations. Component wave delineation has been tried both in time and frequency domains, but time domain techniques seem to provide superior performance. It is well known that the spectra of the component waves such as P, QRS and T waves in the ECG signal overlap with each other and as such cannot be recovered by simple filtering [Murthy and Niranjan, 1992].

Greenhut *et al.*, [1989], used the convexity operator to distinguish between the sharp and high amplitude waveforms (QRS complex) and slowly changing and low amplitude waveforms. Convexity is a parameter intrinsically related to the slope and curvature of the waves. In the same work, the J point was detected using the tangent operator, another parameter related to the slope. Trahanias and Skordalakis, [1989], reduced the ECG signal into a set of peak and segment patterns. Though the approach is similar to syntactic pattern recognition, it is presented in a way that can be followed without difficulty by non-specialists in the field; *i.e.*, no background in formal grammars or parsers, *etc.*, is required.

McLaughlin *et al.*, [1993], discuss the topic of the choice of end points for T waves. The algorithms investigated the detection of T wave end using a) the intercept of the iso-electric line and maximum T wave slope, thresholds applied to b) the T wave and c) the differential of the T wave. According to the study, the technique of calculating the differential of the T wave and comparing with a threshold works better than the other two methods.

The ST-segment in the ECG is defined as the portion between the end of the QRS-complex and the beginning of the T wave. The J point usually indicates the end of the QRS complex. A robust method of estimation of ST segment boundaries provided by Suzuki and Ono, [1992], by entirely using Adaptive Resonant Theory network, since they felt the conventional algorithms making use of thresholds lack the robustness. Similarly, Rasiah *et al.*, [1995], provided an adaptive method using backpropagation algorithm to detect QRS complexes. Though such methods can be combined to form an entirely neural network based feature extraction process, design and performance of the neural networks need to be standardized for tasks other than beat and rhythm classification.

Peter Bonadonna, [1998], in his article on understanding QT/QTc measurements, provides a QT chart and a QT duration graph showing the variation of QT interval with respect to RR interval values. Such a detailed study provides enough information in modeling an ECG beat. Murthy and Niranjan, [1992], suggested a solution in frequency domain by filtering the discrete Fourier transform of the signal. The sets of samples in each component wave are transformed into a complex sequence with distinct frequency band. Multiplication of the transformed signal with a complex

sinusoidal function allows the use of a bank of low-pass filters for the delineation of all component waves. Still, normalized mean square error for the whole beat seems to be above 15%.

A combination of ideas is needed to overcome inherent problems like non-linearity, non-stationarity, missing beats, noise, *etc.*, in detecting fiducial points. Filtering and signal pre-processing alone may not provide an ideal solution for tackling the problems, and ideas from mathematical morphology, geometry, and rhythm information are all needed for an ideal fiducial point detector. The present work attempts to combine these ideas, especially for short-term, routine clinical data of around six seconds duration and the design is expandable for complete Type A and B classifications involving multi-lead, long-term data.

## 2.6. Feature Set - Is there a Standard?

Starting with original work of Pipberger, [1961], many modifications have gone into the selection of features. Most of the initial researchers chose to apply the visual features available in a routine ECG beat. Features like RR interval, PR interval, QT interval and ST slope and QRS slope, P wave amplitude, QRS amplitude and T wave amplitude all provided vital information regarding various rhythm disturbances. Application of raw ECG as input for pattern recognition is not possible with syntactic and statistical methods, but neural networks can handle it. But, even for beat-by-beat analysis, or for QRS classification, the computation and training data needed are very high for raw data [Xue *et al.*, 1992]. The more important problem is, however due to non-stationarity constraints, which prevent proper alignment of different, beats for classification [Lee, 1989]. Automatically modifying the structure of neural network

depending on the present input is a more complex one and practically impossible with present technology.

The role of features becomes vital in the above context. In ECG interpretation, a feature set of size 10-20 values, consisting of intervals and amplitudes and quantities derived from them, is found to provide as much information as an entire beat of raw signal [Borovsky and Zywietz, 1980; Garfein, 1990]. The end-users, physicians and cardiologists, have also been trained to think in terms of features with the last four decades of computerization effort. This fact is even more accurate for syntactic and statistical algorithms for the automated ECG interpretation process [Marquette ECG Machines, 1999]. Another advantage of feature extraction occurs with respect to noise in observed beats. Impact of noise is relatively easier to reduce in initial fiducial point detection stage than in final classification step. Computational hazards can be avoided if the noisy beats are detected earliest in the classification system.

All algorithms available in CSE study and elsewhere use feature extraction as their first task before pattern classification [Willems *et al.*, 1990]. Better techniques using digital signal-processing methodologies have improved the feature calculation to nearly a perfect task. The major gain from the feature extraction algorithms is the elimination of non-stationarity characteristics. The problem changes from one of stochastic non-stationary process to one involving normally distributed random variables. The distribution of variables indicates their wide-sense stationary nature [Elghazzawi and Geheb, 1997]. A classification task, though non-linearly separable, becomes simpler in terms of dimensions of feature space and computation time involved. Feature space reduction is reflected as a decrease in number of training data.

Elimination of unnecessary features has always been important in ECG interpretation [Baxt, 1992]. Even minor ECG variations seemed to carry classification information according to Baxt, and his attempts to choose a smaller set of input features always provided inferior performance.

The application of newer features like linear predictive coefficients, *etc.*, may be good when used independently but cannot become standards unless rigorous evaluation criteria are applied. The other factor to consider is with respect to traditional features, well known to the end-users, like ST segment slope, PR interval, *etc.*, that cannot be neglected. Hence, a compromise has to be reached in choosing parameters based on familiarity, input space, computational complexity and fit into the overall scheme.

## 2.7. Frequency Domain Features

This section presents an overview of frequency analysis of signals and targets especially ECG signals, which is of importance in the present context. Developments in spectral estimation related to both non-parametric and parametric methods are described.

### 2.7.1. Spectral Estimation – Brief History

Spectral estimation has its roots in ancient times, with the determination of the length of the day, the phases of the moon, and the length of the year. In an interesting and earlier work, Schuster [1898], distinguished two types of periodicity: obvious periodicity and hidden periodicity. Obvious periodicity is visible as soon as a sufficient record has been reached. A relevant example is beat periodicity of ECG signal (with non-stationarity). However, hidden periodicity is hidden behind irregular fluctuations,

and the detection and estimation becomes difficult. Added to the problem is the unknown nature of the periodic function. Such a problem exists in ECG frequency analysis also. Schuster is also credited with having coined the term "periodogram". He related the resolving power in frequency domain to the period T.

Modern spectral estimation began with the breakthrough for the analysis of short time series made by Tukey in 1949. This work led to the high growth of spectral analysis techniques, in many application areas. Still, the computations proved expensive. The next breakthrough occurred with the discovery of Fast Fourier Transform (FFT) in 1965 independently by Tukey and Cooley and by Gordon Sande. This development, along with silicon chip technology, has brought spectrum analysis to bear on a wide range of problems. Another breakthrough occurred with the introduction of maximum entropy methods into spectrum analysis by John Burg in 1967 [Robinson.E.A., 1982].

2.7.2. Classical Methods

There is a clear lack of an accepted definition of optimality in the comparison of spectral estimators. The oldest method of spectral estimation was called periodogram and was proposed by Schuster in 1898. The periodogram estimator is defined as

$$\hat{P}_{PER}(f) = \frac{1}{N}\left|\sum_{n=0}^{N-1} x[n]\exp(-j2\pi fn)\right| \tag{2.1}$$

It might be supposed that if enough data are available, say N→ ∞, then

$$\hat{P}_{PER}(f) \rightarrow P_{\infty}(f) \tag{2.2}$$

or that the periodogram is a consistent estimator of the PSD. But, it has been found that though the mean converges to true PSD as N→∞, the variance does not tend to zero as

$N \to \infty$. Instead, it is observed that the variance is a constant independent of N and hence unreliable. To overcome this problem, another approach is to average the periodograms of K independent data records, all being realizations of the same random process and of the same length $0 \le n \le L-1$. Then the *averaged periodogram estimator* is defined as

$$\hat{P}_{AVPER}(f) = \frac{1}{K}\sum_{m=0}^{K-1} \hat{P}_{PER}^{(m)}(f) \tag{2.3}$$

However, the averaged periodogram shows an increase in amount of bias over that of the periodogram and another problem is the availability of a huge data set. Zero padding is another vital concept in much spectral estimation. For example, when the data length is small, with zero padding, the effective data set becomes

$$x'[n] = \begin{cases} x[n] \text{ for } n = 0,1,\ldots\ldots,N-1 \\ 0 \quad \text{for } n = N, N+1,\ldots\ldots,N'-1 \end{cases} \tag{2.4}$$

The frequency spacing will then be $1/N' < 1/N$. *No extra resolution is afforded by zero padding but only a better evaluation of the periodogram.* Hence, absence of huge data and non-stationarity characteristics make the spectral analysis of ECG beats using periodogram technique very difficult [Voss, 1992; Robinson, 1982].

As suggested by Kay, [1992], the poor performance of the periodogram can be explained by poor performance of autocorrelation function (ACF) estimator, since the equivalent form of the periodogram is given by

$$\hat{P}_{PER}(f) = \sum_{k=-(N-1)}^{N-1} \hat{r}_{xx}[k]\exp(-j2\pi fk) \tag{2.5}$$

where,

$$\hat{r}_{xx}[k] = \begin{cases} \frac{1}{N}\sum_{n=0}^{N-1-k} x^*[n]x[n+k] & \text{for } k = 0,1,\ldots\ldots,N\text{-}1 \\ r^*_{xx}[-k] & \text{for } k = -(N\text{-}1),-(N\text{-}2),\ldots\ldots,-1 \end{cases} \tag{2.6}$$

The poorer estimates of the ACF at higher lags are a result of the fewer number of lag products inside the averaging function.

The popular spectral estimation method according to Blackman-Tukey tries to overcome the above problem by weighting the ACF estimates at higher lags less or to use the spectral estimator

$$\hat{P}_{BT}(f) = \sum_{k=-(N-1)}^{N-1} w[k]r_{xx}[k]\exp(-j2\pi fk) \tag{2.7}$$

where w[k] is a real sequence and is called a *lag window* with the following properties:

1. $0 \leq w[k] \leq w[0] = 1$
2. $w[-k] = w[k]$
3. $w[k] = 0$ for $|k| > M$

where $M \leq N\text{-}1$. A major problem with the Blackman-Tukey estimation method is that it produces rather broadband peaks from short recordings (in the present case about 200 samples). Another major drawback is with respect to inherent inability to accommodate non-stationarity very well. This factor becomes even more obvious for beats with varying lengths. Variable zero padding has been applied by some authors, but the results are stable for longer data length only [Thakor *et al.*, 1984].

### 2.7.3. Need for Parametric Modeling

The classical spectral estimation methods make use of the second-order statistics of a random process in terms of either the auto-correlation function (ACF) or the power

spectral density (PSD). Both are non-parametric descriptions. The classical methods used Fourier transform operations on either windowed data or windowed ACF estimates. Windowing of data or ACF values makes the implicit assumption that the unobserved data outside the window are zero, which is an unrealistic assumption. A smeared spectral estimate is the consequence of windowing. Using prior information may permit the selection of an exact model for the process that generated the data samples. With the knowledge of prior information, the need for windowing can be eliminated and results in a dramatic improvement over the conventional FFT spectral estimator, especially for short data record [Kay, 1992; Marple, 1987]. These factors make a parametric model based spectral estimation attractive for estimating the spectrum of an ECG beat.

2.7.4. Different Parametric Methods

Many discrete-time random processes encountered in practice are well approximated by a *time series* or *rational transfer function* model. In this model, an input driving sequence u[n] and the output sequence or observed data x[n] are related by the linear difference equation,

$$x[n] = -\sum_{k=1}^{p} a[k]x[n-k] + \sum_{k=0}^{q} b[k]u[n-k] \tag{2.8}$$

This most general linear model is termed an auto-regressive moving average (ARMA) model. The ARMA model noise u[n] is not an additive or observation noise, as encountered in signal processing applications. u[n] is an innate part of the model and takes care of the random nature of the observed process x[n], the ECG beat in the

present case. The system function H(z) between the input u[n] and the output x[n] for the ARMA process (2.9) is the rational function

$$H(z) = \frac{B(z)}{A(z)} \tag{2.9}$$

Where, A(z) = z-transform of AR branch = $\sum_{k=0}^{p} a[k]z^{-k}$ and

B(z) = z-transform of MA branch = $\sum_{k=0}^{q} b[k]z^{-k}$

Assumption of H(z) to be a stable and causal filter makes the equation (2.9), a valid description of a Wide Sense Stationary (WSS) random process [Marple, 1987].

The power spectral density of the ARMA output process becomes

$$P_{ARMA}(f) = P_{xx}(f) = \sigma^2 \left| \frac{B(f)}{A(f)} \right|^2 \tag{2.10}$$

Where, A(f) denotes A(exp[$j2\pi f$]) and B(f) denotes B(exp[$j2\pi f$]). This complete model is sometimes referred to as a pole-zero model and is denoted as an ARMA(p,q) process. If all the a[k] coefficients except a[0]=1 vanish for the ARMA parameters, then

$$x[n] = \sum_{k=0}^{q} b[k]u[n-k] \tag{2.11}$$

and the process is strictly an moving average (MA) process of order q, and the PSD is

$$P_{MA}(f) = \sigma^2 |B(f)|^2 \tag{2.12}$$

If all the b[k] coefficients except b[0]=1 are zero in the ARMA model, then

$$x[n] = -\sum_{k=1}^{p} a[k]x[n-k] + u[n] \tag{2.13}$$

and the process is strictly an auto-regressive (AR) process of order p. The PSD is denoted by

$$P_{AR}(f) = \frac{\sigma^2}{|A(f)|^2} \tag{2.14}$$

This model is termed an *all-pole* model and is indicated as an AR($p$) process. Theoretically, an AR($p$) process can approximate any MA or ARMA processes, with a higher value of $p$. In the present work, a combination approach is used to arrive at a feature space, which adapts to non-stationarities as well as providing good classification advantages over any single approach.

2.7.5. Applications in Frequency Domain related to ECG signals

Applications related to ECG signals in frequency domain have used Fourier-transform based estimation methods as well as maximum entropy based methods. Recently, high frequency analysis related to ventricular late potentials has become prominent [Voss *et al.*, 1992; Berbari *et al.*, 1990]. Original definition of Blackman and Tukey produces rather broadband peaks for short-term recordings, due to the limited time-width [Voss *et al.*, 199]; Kay, 1992]. This observation is relevant for short-term clinical ECG data. Thakor *et al.*, [1984], studied the averaged frequency spectrum of ECG signal and attempted to study the averaged spectrum for individual components also. A plot of the spectrum is shown in Fig. 2.9. The spectral estimation was based on more than 150 beats and used Fourier transform based methods. The figure indicates that the frequency content is important above about 1 Hz. However, baseline seems to be affected greatly in the absence of low-frequency components. In the present work, a frequency band of (0.05-100) Hz is used for clinical data and (0.1-100) Hz is used for MIT-BIH data.

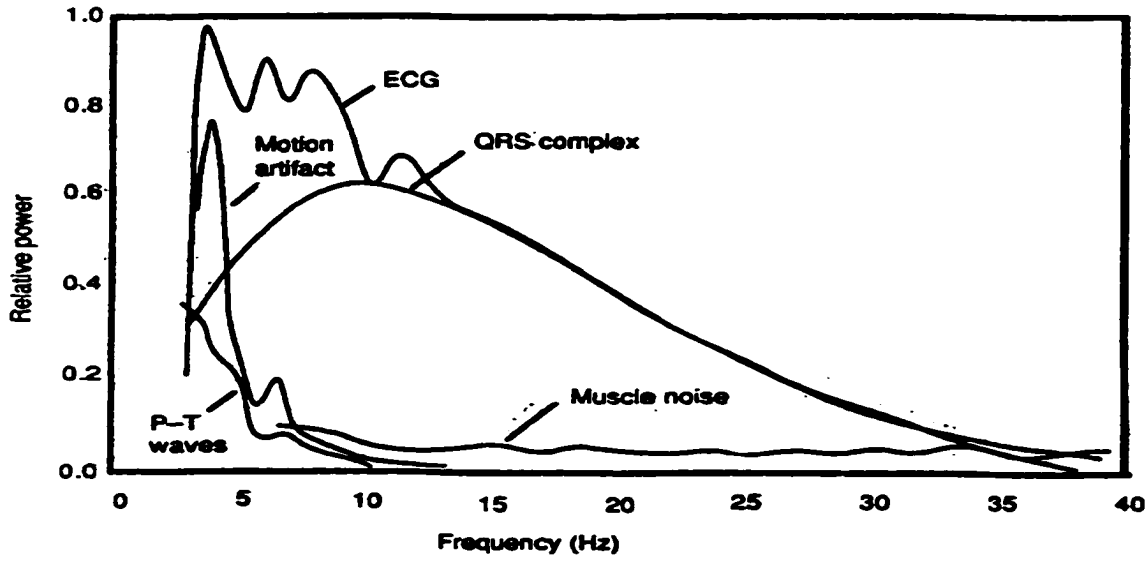



Fig. 2.9. Relative power spectral densities of ECG components averaged over 150 beats [Thakor *et al.*, 1984]

In their comparison of four different techniques for recognition of ventricular fibrillation as well as atrial flutter, Clayton *et al.*, [1993], have advocated the higher amount of information provided by the signal spectra, compared with other time domain methods and methods based on auto-correlation calculations. They have suggested the importance of combining spectral algorithms with an efficient QRS detector and activating the spectral domain detection only if no QRS complex can be detected. This concept is vital in the current work and eliminates the necessity for excessive calculations.

Recent trends point to more direct application of frequency domain analysis related to individual waves in ECG beats. Langberg *et al.*, [1998], presented a new technique to analyze atrial fibrillation. The major steps in the procedure are band-pass-filtering, subtraction of QRST segment, and Fourier transformation. The study suggests that the frequency analysis of the surface ECG may be a useful means to augment

clinical decision-making. In another related work, Strohmenger *et al.*, [1997] used parameters like median frequency, dominant frequency and amplitude for predicting the success of counter-shocks in case of ventricular fibrillation. Counter-shock success is defined as a supra-ventricular rhythm after a counter-shock.

In recent years, more effort is being spent on converting the older algorithms to newer microprocessor based functions like arm-lead reversal detection, optimal trace positioning, lead-off and/or artifact detection, sensitivity detection, drift reduction and AC interference rejection [Marquette MAC ECG Analysis System]. Commercial systems are getting refined and becoming more reliable because of these efforts. Evaluation results of these new implementations under commercial settings are not available in public domain.

## 2.8. Neural Network based ECG Pattern Recognition

In the last decade, neural networks regained their popularity with the ease of implementation and several successful implementations of pattern-recognition problems [Maren *et al.*, 1990]. In last decade, all such developments became feasible with the surge in popularity of back-propagation learning algorithms, independently suggested by Werbos, [1974], as well as Rumelhart, [1986]. The theory behind adaptive resonant theory (ART) networks also became popular during that time and resulted in applications related to ECG interpretation and data compression [Han, 1993; Saxena *et al.*, 1997]. As noted by Casy Klimasauskas, a co-founder of NeuralWare, Inc., in 1992, many of the neural network applications in modeling, as well as in classification, are the results of replacing existing statistical, modeling, or signal processing techniques with

neural computing techniques. Similarities to other modeling and function approximation technologies reduce the costs and risks associated with deploying it.

Soon after, electrocardiogram interpretation problems attracted the attention of neural network experts. Initial efforts focused more on the technology, rather than the problem. Even raw signals were tried as inputs in a few networks. The significance of feature extraction was not realized for some time. In the early 1990's, several specific implementations of automated ECG interpretation projects based on ANN technology were published [Baxt, 1992; MacFarlane, 1993].

Presently, at least 10-15 groups are working on applying and perfecting ANN technology in ECG problems. In last few years, implementation of new structures as well as applications related to other signals have been explored. According to Levary, [1993], a combination of the capabilities of artificial neural networks with the rule-based systems results in a system which will deal very well with both details as well as patterns. In the overall context of computerized ECG interpretation, such a combination network can accommodate the addition of clinical details easily.

### 2.8.1. What are Neural Networks?

Neural networks are one of the successful attempts in building machines that learn by experience. A neural network may be defined as a massively parallel distributed processor that has a natural propensity for storing experimental knowledge and making it available for use. Presently, artificial neural networks (ANNs) do not approach the complexity of brain. However, two similarities exist between the brains and ANNs. The fundamental unit of both ANNs and brains is a neuron and neurons behave as functions. A model of a sample neuron is shown in Fig. 2.10. (a).

[Wasserman, 1989] and relations between different sample transduction functions are also shown in Fig. 2.10. (b). The other similarity is the nature of connections, which are massively parallel.

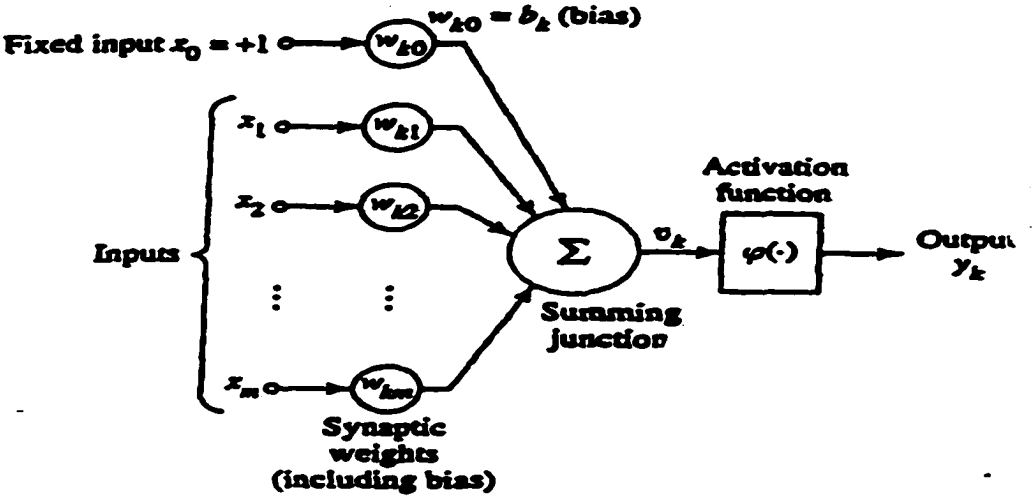



Fig. 2.10. (a) Model of a sample neuron

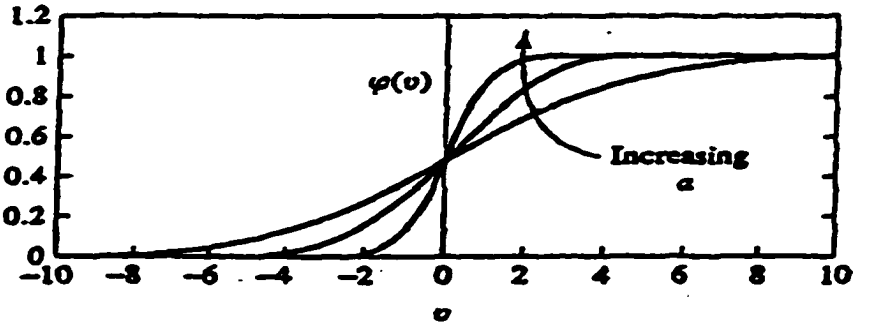



Fig. 2.10. (b) Relation between different transduction functions

2.8.2. Properties and Types of Neural Networks

Artificial Neural Networks (ANNs) are formed by the interconnection of neurons. A single layered neural network is the simple perceptron. The interconnections might be different in one of the following two ways;

(i) Neuron operating characteristics, and

(ii) Different connecting patterns.

The simplest type of ANN is a multilayer feedforward network. This type of connections of neurons overcomes the inability of the individual perceptrons to classify anything apart from linearly separable variables. The structure of a simple multilayer network is represented in Fig. 2.11. According to Kolmogorov's theorem, two layers of processing units are sufficient for carrying out any kind of linear demarcation required [Hornik.K, 1991].

The backpropagation (BP) algorithm is the most widely used learning algorithm for the feedforward networks. The advent of the BP learning rule has provided an impetus in applications using multilayered feed-forward networks [Maren, 1990]. Backpropagation enabled the errors in outputs to propagate through the inner layers, which enabled the multilayer feed-forward networks to adjust their weights based on training.

The layers in between the input and the output layers constitute hidden layers and neurons in those layers are referred as hidden neurons. There are one or more hidden layers in multilayered feedforward networks. The function of hidden neurons is to intervene between the external input and the network output. By adding one or two hidden layers, the network is enabled to extract higher-order statistics, by virtue of the extra set of synaptic connections and the extra dimension of neural interactions [Churchland and Sejnowski, 1992]. Optimal design of a neural network for a particular task has not been standardized yet and so far the implementations have been made on previous experiences and trial and error methods [Xue and Tompkins, 1992].

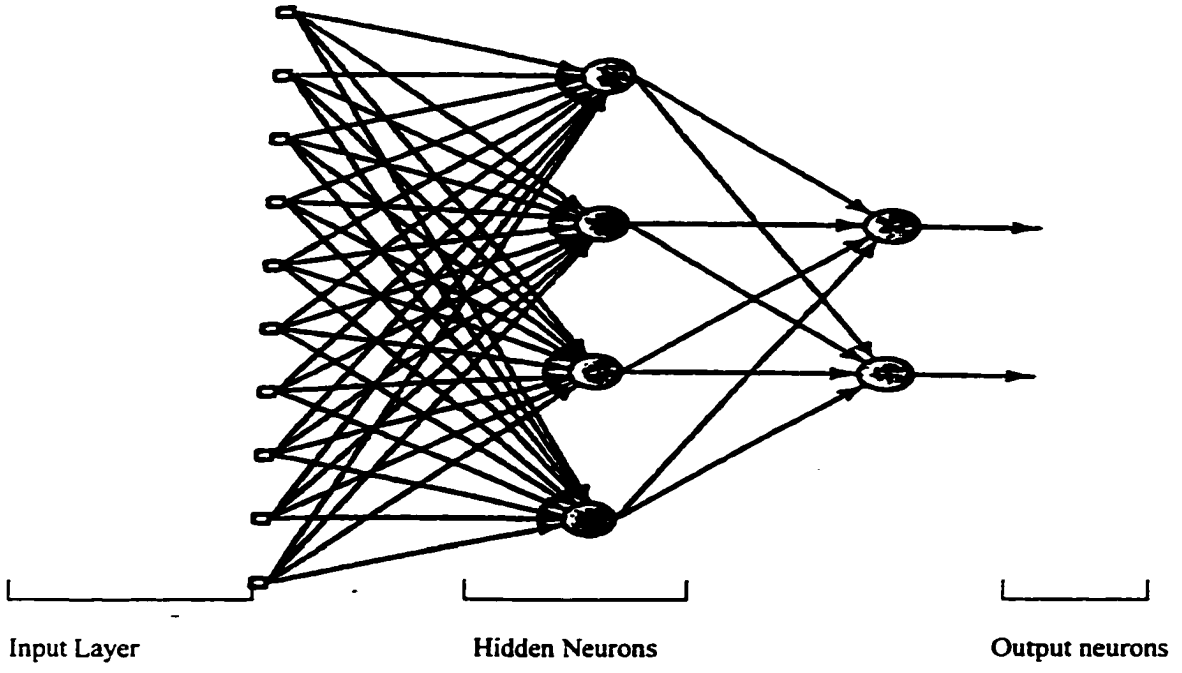



Fig. 2.11. Structure of a simple multilayer network

2.8.3. Comparison with Expert Systems

In the last four decades, Artificial Intelligence grew into two major approaches, namely, expert systems and neural networks. The main goal for both the methodologies is to replicate the function of brain, using the computing power of machines. As noted by Zahedi, [1991], the expert system approach assumes the brain as a black box and imitates the human reasoning process. It processes knowledge sequentially, represents it explicitly, and mostly uses deductive reasoning. Learning takes place outside the system. On the other hand, neural networks treat the brain as a white box and imitate its structure and function, using a parallel approach to simulate human intelligence. Artificial neural networks represent the knowledge implicitly within its structure, and apply inductive reasoning to process knowledge. Learning takes place within the system. In ECG interpretation problems, a combination of principles derived from

ANNs and expert systems suggests better results [Kuppuraj, 1995]. Artificial neural networks, once properly trained, seem to provide results considerably faster than statistical procedures, which are in turn faster than syntactic methods, by a factor of around two [MacFarlane and Lawrie, 1974; Xue and Tompkins, 1992].

2.8.4. ANNs as ECG Classifiers

Baxt, [1992], and Baxt and Skora, [1996], used an ANN to identify the presence of myocardial infarction and analyzed the clinical variables along with ECG findings. The study is an example of research effort possible in a huge clinical setting at University of California, San Diego Medical Center. The study involved 706 patients presenting with anterior chest pain. This study was to realize the significance of clinical variables driving decision-making, which is possible with step-by-step perturbation of the network. Final results suggested that the network comes to clinical closure based on the settings of all variables in a pattern and that the impact of a single variable cannot be taken out of the context of a pattern. As expected, significant ischemic ECG changes affect the decision-making process more than the other clinical details. The diagnostic network is of the structure 20×10×10×1. In their later results, the performance of ANN seems to be better or comparable to that of physicians [Baxt and Skora, 1996].

In his doctoral dissertation work, Ohlsson, [1995], discusses in detail the implementation strategies for the problem of detecting lead reversals in left/right arm positions. A typical methodology is discussed as a model for an ANN-based ECG interpretation system with a large data set for training and testing. However only the cardiologists know the existence of such problems. In another typical clinical implementation, Heden *et al.*, [1996], explain the strategy for classifying the probability

of healed myocardial infarction using ANNs. The study details the selection of features, input, hidden, and output layer size and the data needed for training and testing.

In a work connected to the above dissertations, Heden *et al.*, [1996] have compared the performance of ANNs with that of an expert electrocardiographer, and the results showed a great agreement in sensitivity and selectivity. The outputs of ANN were transformed into verbal statements. Electrocardiogram features included for this study were Q, R, and S wave amplitudes; Q and R wave duration and three amplitudes within the ST-T segment. The above features were calculated in leads $V_2$, $V_3$ and $V_4$.

Devine and MacFarlane, [1993], developed both the signal processing and their own ANN, to detect the ST-T abnormality. Inputs included eight interval amplitudes between S and T points. Neural network performance is compared with the performance of conventional criteria. The network had a higher overall sensitivity (97% against 89%) and specificity (88% against 82%) compared to conventional rule-based criteria.

Silipo *et al.*, [1995], used a combination of two databases, one from the European Society of Cardiology and the other from the MIT-BIH database. A comparison of performance between Threshold Crossings vs. ANN structure was done. Input was the 150 msec of data around QRS complex. Input values range between [-0.5, 0.5]. When raw ECG data are used for pattern recognition, due to QRS shape overlapping, the training set is limited for each subject. When the number of classes increased, the analysis difficulty increased because one network alone is unable to transfer to the weights the information of too many features. Multiple networks may be

needed for learning the characteristics of very widely varying classes. The same problem exists in other neural network architectures like adaptive resonance theory mapping (ARTMAP). This problem restricted Han, [1993], to classifying only normal and PVC beats.

Kuppuraj, [1995], implemented a combination approach involving both quantitative and qualitative information for the detection of arrhythmia. The approach was bound to provide better performance; however, the system CORDOC did not include the feature extraction steps and suffered from non-stationarity problems. The feature extraction step is bound to decrease the problem-space dimension as well as completely - eliminating the non-stationarity problems associated with ECG interpretations. Training complexity increases with the size of input space and results in the requirement of large data set [Haykin, 1999]. The solution to this problem consists of one of the two methods as far as ECG signals is concerned; the first option is to make use of the numerical features available in commercial ECG machines as done by many clinical researchers [Heden *et al.*, 1997]. However, these features may not be exactly suited for neural network implementations. They do not include frequency domain features and also the features are machine specific. The other option, as followed in present work, is to optimize the entire process, by extracting features from the raw signal itself.

## 2.9. Conclusions

The detailed background material covered in the present chapter helps the present work to form a definitive methodology, as indicated in the next chapter. Automated ECG interpretation derives ideas from a variety of new and old areas and a

combination approach is more likely to succeed. The next chapter provides a detailed account of the research design in terms of overall system design as well as minor details in the form of formula and algorithms. A detailed literature survey revealed the advantages of starting at the sub-system level and perfecting their performances. The motivation for the concept of bottom-up design is the freedom it provides in the choice of features and the resulting control on the whole process. An effort has been made to present finer points in implementation often neglected in similar clinical studies.

# CHAPTER 3

# DESIGN AND METHODOLOGY

## 3.1. Task Description

The schematic of the entire methodology is shown in Fig. 3.1. The uniqueness of the implementation results from need-based use of required algorithms, as indicated by triggers in various sub-systems. The overall goal of the present thesis is to design an automated single-lead (lead II) ECG classification system, which satisfies the following performance criteria:

(i) Optimized design implementing need-based feature extraction and pattern recognition.

(ii) An ability to identify the twelve most prevalent ECG beat abnormalities correctly.

(iii) An ability to identify complex or irregular data, before utilizing much computational resources.

(iv) An ability to be extended to a complete multi-lead ECG interpretation system.

In the present context, an optimized design is defined from the point of view of interaction between the blocks shown in Fig. 3.1., rather than the context of software and hardware details [Schimminger, 1998]. Each sub-system is chosen based on two factors; overall compatibility with other sub-systems, and its performance characteristics. Performance analyses are rigorous in this life-saving diagnostic procedure, and extensive literature is available regarding the minimal standards

[Moody, 1992; Laguna *et al.*, 1997]. The committee on Common Standards for Quantitative Electrocardiography (CSE) provides a detailed survey of the algorithms and equipment characteristics [Leblanc, 1989; Willems *et al.*, 1990]. Based on their day-to-day application of the procedures, clinicians and cardiologists also voice their own concern at the performance of the ECG interpretation machines [Sheffield, 1987]. Such performance indices are taken into consideration in present work, and they have an impact on design of all the sub-systems shown.

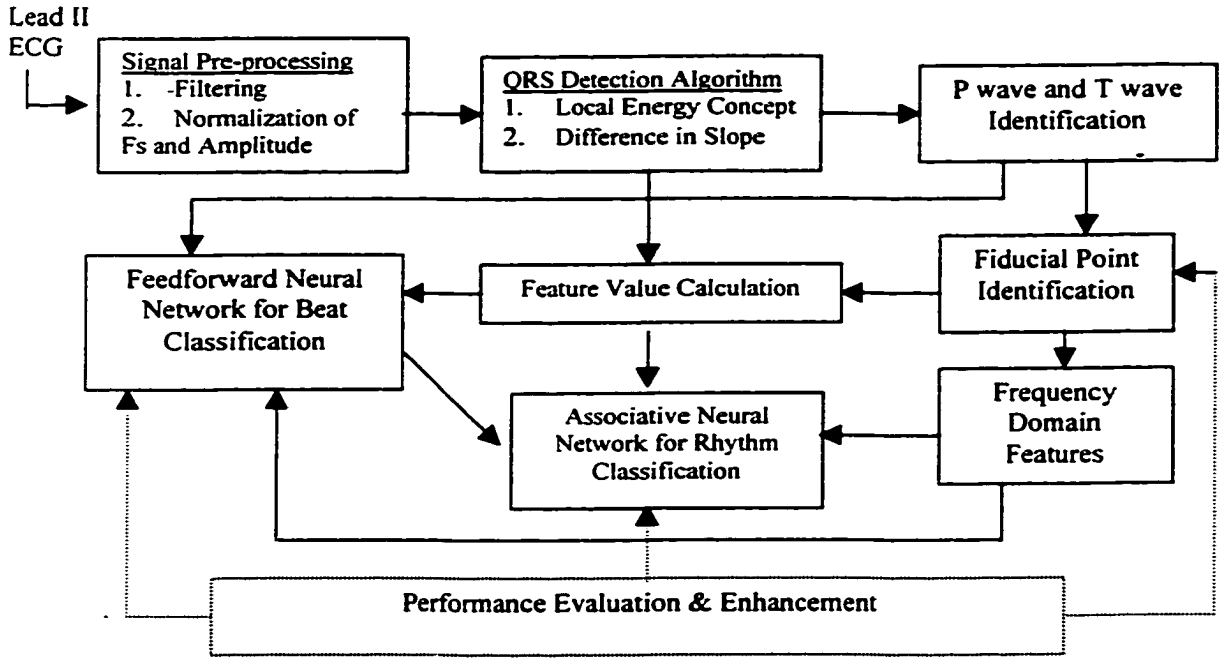



Fig.3.1. Block diagram representation of the overall ECG interpretation - preprocessor

The proposed work is typically defined under Type-C class of systems, which provide structural information about ECG beat and rhythm, including ST segment aberrations, presence of artifacts, noise, *etc.* Development of Type-C systems needs short-term ECG data for training and testing [Moody, 1992; Sommo *et al.*, 1980].

## 3.2. Pre-processing

Clinical ECG data were acquired from limb lead II using a high-resolution acquisition system designed using INA-104 instrumentation amplifiers. Electrocardiogram data were acquired from infarct subjects both in intensive coronary care unit and out-patient ward in Kilpauk Medical College Hospital, Chennai, India. A PCL-206 based analog-to-digital converter (ADC) performed analog-to-digital conversion, with a resolution of 12 bits, and an IBM PC-486 interface card. Digital data were acquired at a sampling rate of Fs=250 Hz. The analog filter range for clinical data was (0-100) Hz and only a low-pass filter of $f_c$=100 Hz is used.

The sampling frequency of digital ECG data in the MIT-BIH database is 360 Hz, with an analog passband from 0.1-100 Hz relative to real time. However, the sampling frequencies of the clinical ECG from other sources vary from 250 Hz to 500 Hz. Normalization techniques are needed in the pre-processing stage since ECG data in present work are derived from sources differing in their analog-to-digital converter characteristics and sampling frequencies. Such a problem is expected in any multi-clinic trial with a huge data requirement. Normalizing with respect to amplitude is easier to implement, but sampling frequency normalization for data from different sources is more difficult.

Therefore, a digital low-pass filter of $f_c$=100 Hz is applied for all the data. This filter is designed with Matlab® package and belongs to the IIR class filter with Chebyshev type-I of sixth order. Since the analog filter cutoff is only 100 Hz, the information lost is minimal for the present task. An interesting aspect about the frequency range is that even for machines with a higher low-pass cutoff, final detection

of P wave and T wave boundaries usually involves a smoothing process, resulting in a lower cutoff [Greenhut *et al.*, 1989; Skordalakis, 1986; McLaughlin *et al.*, 1993]. After the low-pass filter, the mean of the signal is subtracted from the original signal. This de-trending result in the removal of the zero Hz component and base-line returns to an amplitude of zero. Hence, the final choice of ECG bandwidth is (0.1-100) Hz for MIT-BIH data and (0.04-100) Hz for the clinical data. This frequency range avoids disturbing minor variations in base-line and ST segments and allows the regular frequency range of P and T waves and affects only a little of QRS complex.

A methodology has been developed by Srikanth *et al.*, 1998, as part of the present dissertation for sampling rate conversion from the original sampling frequencies of 360 Hz and 250 Hz *etc.*, to a final sampling frequency of 500 Hz. Eight alternative methods were compared with respect to the same parameters. Cubic interpolation provided very good accuracy in re-sampling, compared to other interpolators based on filter design techniques. Another advantage of cubic interpolation is the uniformly high performance for both down- and up-sampling of the signal. The cubic interpolation algorithm is used for pre-processing both MIT-BIH data and clinical data in the present work. Cubic interpolation essentially fits the points using a third order polynomial and is defined as a curve between two points [Lancaster and Salkauskas, 1986]. Cubic interpolation belongs to the Lagrange interpolation class and appears to be more accurate compared to linear interpolation. It is simple and fast compared to spline interpolations [Schafer and Rabiner, 1972]. Results of this comparison are detailed in next chapter on results.

Segments of data of six to ten seconds duration. similar to the ones in routine clinical recordings, are chosen for current implementation. Data duration of six seconds satisfies the requirements for even slower beats as well as arrhythmias like bigeminy, trigeminy *etc.* A minimum of three or four complete beats is available for interpretation. The present focus is on six-ten seconds of routine ECG recordings, since the majority of ECG recordings belong to this category.

3.2.1. Annotation of the Data Set

Detection of fiducial points in an ECG beat (P,Qs,Q,R,S,J,and T points) and identification of beat classes for annotation was done manually to facilitate the study. Using the methodology suggested by Laguna *et al.*, 1997, a user-interface was developed in Matlab® environment to help the experts to identify the fiducial points. A sample diagram of the procedure is illustrated in Fig. 3.2. A graphical user-interface using mouse-clicks are enough to choose the needed fiducual points. Fiducial points detected by the experts include on- and off-sets of the various ECG waves, namely P waves, T waves, and QRS complexes. The presence or absence of U waves is noted. The errors made in initial choice can be detected, and new points can be identified using this program.

At least one experienced electrocardiographer (physician or graduate student) analyzed each chosen data segment. Further, all the beats from data segments of MIT-BIH database include beat-by-beat annotations by two cardiologists. Since the purpose of the present work is a type-C classification statement regarding the structure of ECG beats alone, both rhythm variations as well as morphological variations are taken into account. Original MIT-BIH annotations for beat and rhythm classes and the coding

used in the present work are provided in Appendix-B. A few ECG beats are discarded, due to high noise or abnormal value of calculated features. Details regarding the criteria for discarding ECG data and the final data set are explained in chapter 4 on results.

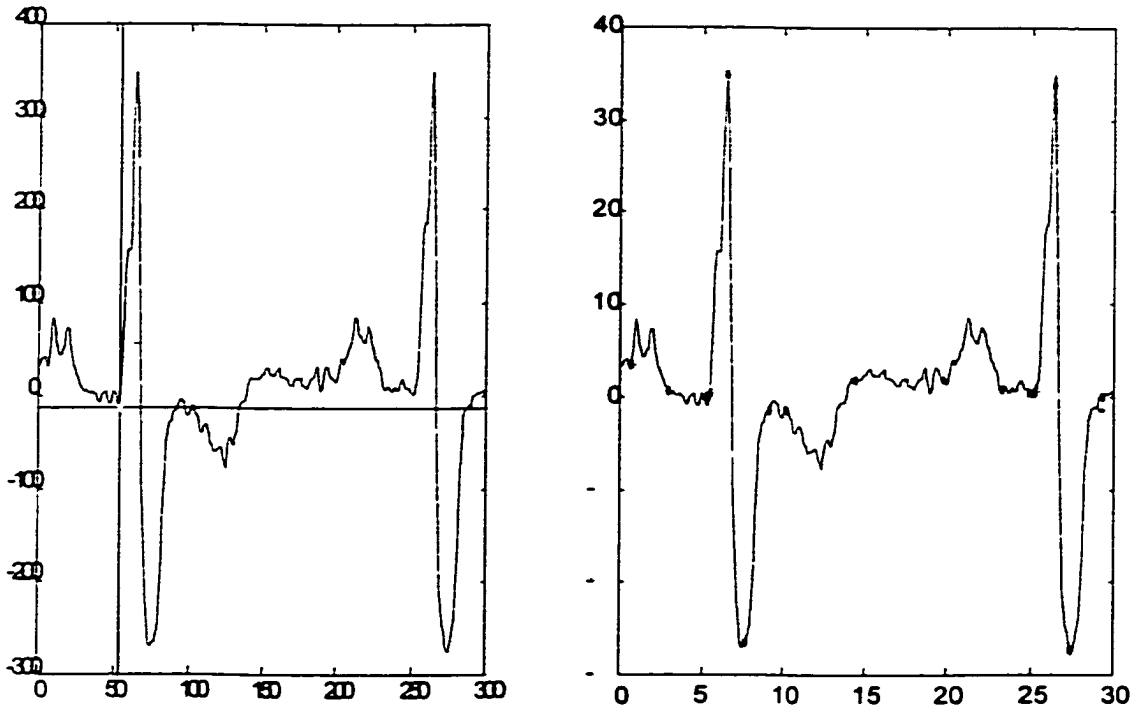



Fig. 3.2. A sample graphical user-interface and the fiducial points identified for expert to select the fiducial points from the screen

3.2.2. Morphological Variations and Arrhythmia Studied

In the present work, normal arrhythmia conditions, and morphological variations like inverted T waves as well as noise, artifacts, *etc.*, are identified. Included ECG beats from MIT-BIH database and clinical recordings mainly fall into one of the following classes; normal (NOR) beats, premature ventricular contraction (PVC), atrial premature contraction (APC), left bundle branch block (LBBB), right bundle branch block (RBBB), nodal (JUN), premature beat (NODE), paced beat (PACE), unclassifiable beat

(Q), ventricular escape beat (VEB), noisy beat (NOI), inverted T-wave (INVT), and ST segment slope (ST), atrial fibrillation (AFIB) and atrial flutter (AFL). A clear definition for each class and a few sample beats are given in Appendix-A. The present work utilizes short-term ECG segments as in routine clinical records. Fig.3.3.(a). indicates a set of normal beats and normal sinus rhythm. Figure 3.3.(b) indicates a collection of normal beats along with a fusion beat and Fig.3.3.(c) indicate a collection of right bundle branch block beats. Figure 3.3.(d) indicate a set of normal and premature ventricular contraction beats and Fig.3.3.(e) indicate a segment with left bundle branch block beats.

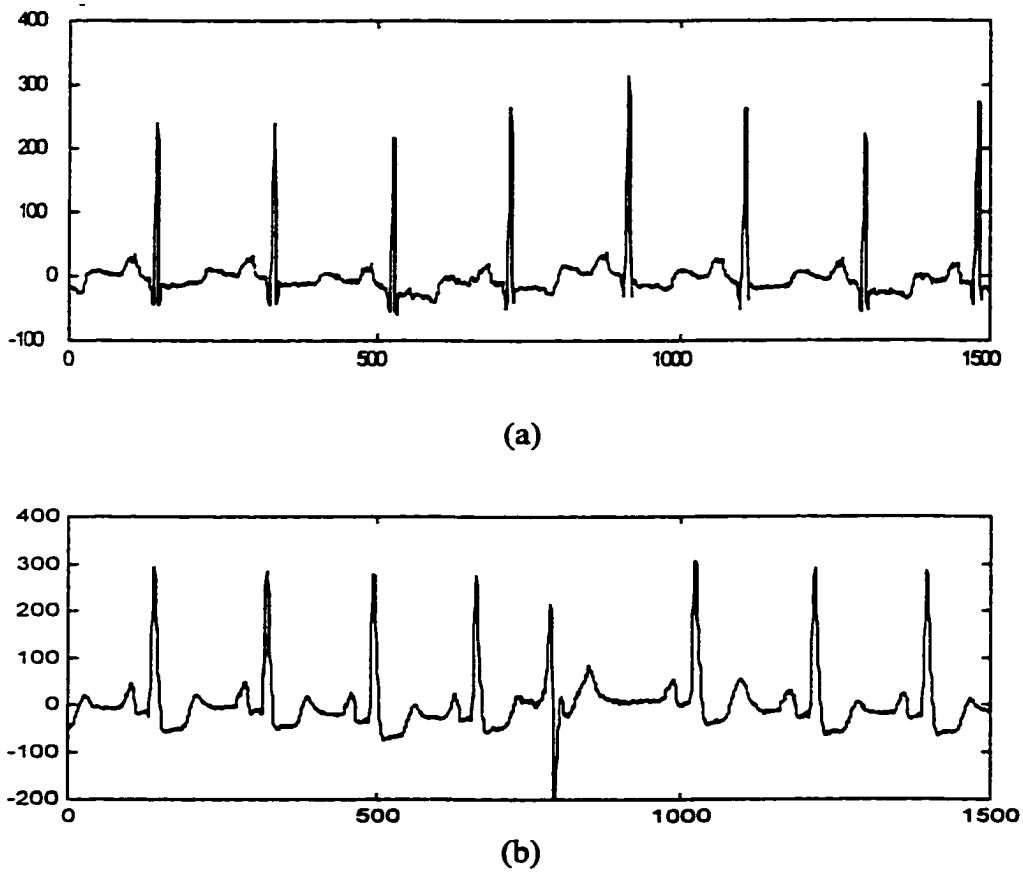



Fig. 3.3. A sample set of lead II ECG segments, belonging to different beat classes

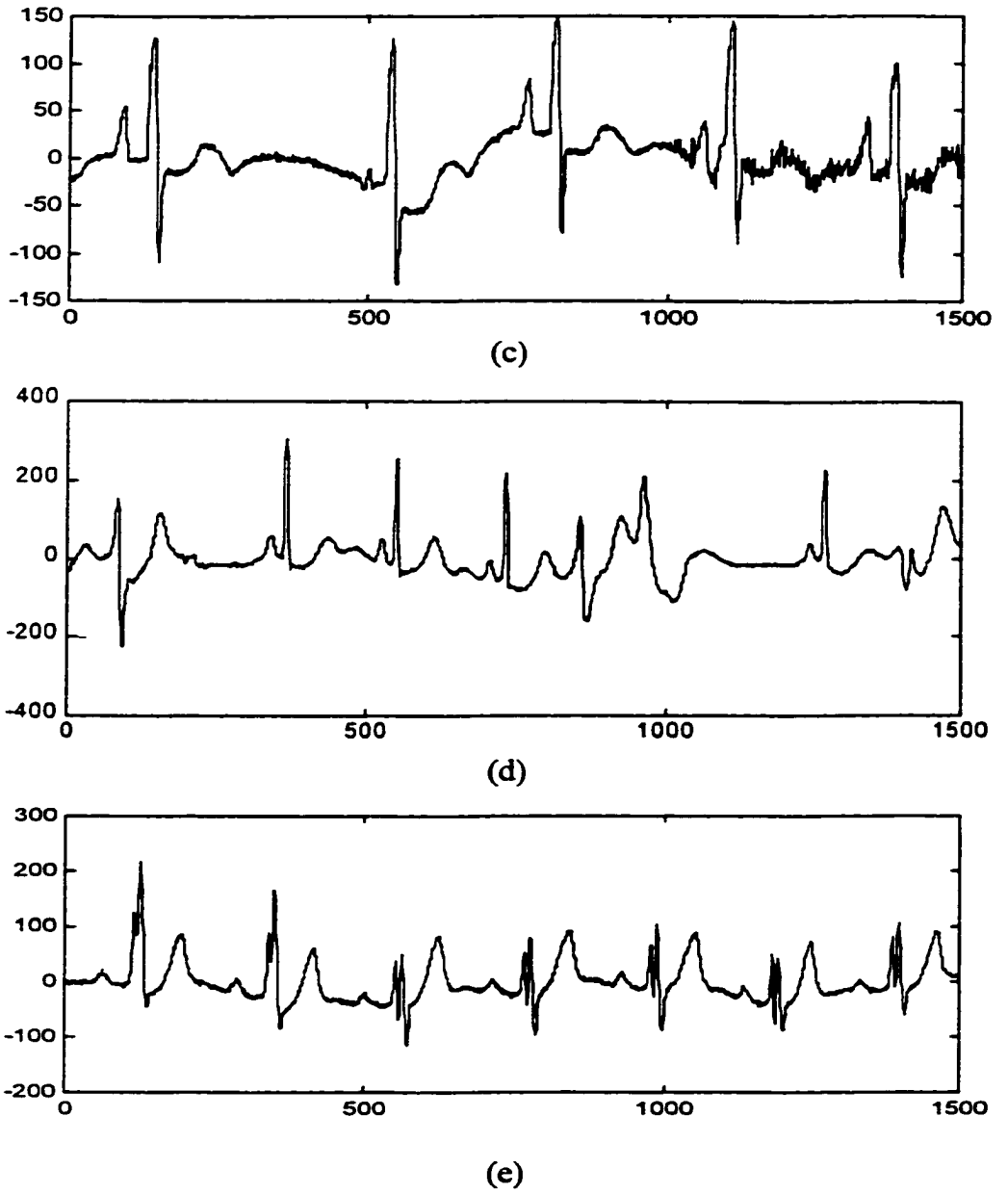



Fig. 3.3. A sample set of lead II ECG segments, belonging to different beat classes (Cont.)

A sample selection of ECG data segments used in analysis is shown in Fig. 3.3. Some data segments may contain beats belonging to more than one class. The present work makes use of the information available in limb lead (II) only. Exact classes finally chosen and classified are presented in later sections and are similar to the types

mentioned above. Beat classification remains the first step before the rhythm annotations. Rhythm annotations like atrial fibrillation, normal sinus rhythm, ventricular bigeminy, ventricular trigeminy, and ventricular tachycardia, *etc.*, indicate characteristics of a group of beats, and are derived from the relations between member beats.

## 3.3. QRS Complex Identification

Precise identification of QRS complexes is the most important task for any fiducial point identification methodology. The R point, the most prominent point in any beat, is first detected inside the QRS complex. Pahlm and Sornmo made a review on QRS detection software in 1984. The classical methods discussed in their work form the basis of many commercial products.

Conceptually, most QRS detectors described in the literature are composed of two entities, a preprocessor and a decision rule. Simple local energy-based QRS detection may be accurate enough for majority of ECG beats as needed in any heart rate variability (HRV) analysis. However, in short-term routine ECG recordings, R-point detection becomes the starting point for all other steps [Srikanth, 1996]. The errors made in R-point detection are more sensitive in a short-term record. The general components in a routine QRS detector are shown in Fig. 3.4. [Pahlm and Sornmo, 1984].

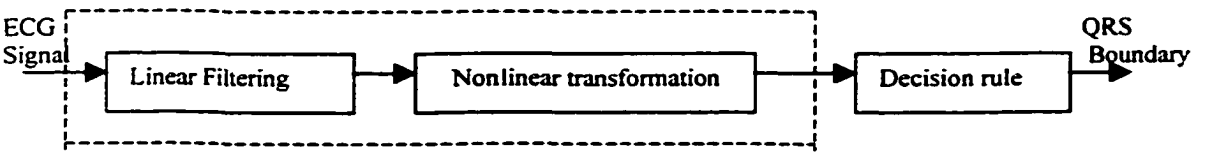



Fig. 3.4. Block diagram of a routine QRS detection algorithm - signal processing approach [Pahlm and Sornmo, 1984]

In the present work, a combination approach is used for the QRS detection and subsequent R-point detection. Apart from pre-processing algorithms, the QRS detection algorithm is the only part common to all input data in the current schematic. Fig. 3.5. indicates the detailed block diagram of the QRS detector implementation. A combination of two different schemes is implemented to cater to all types of QRS complexes. The major algorithm uses short-term energy (or) moving window integrator (MWI). Initial pre-processing for the sub-system involves the high-pass filtering of the signal to highlight the sharp characteristics of QRS complex. Chebyshev type-I IIR filter of sixth order is used for high pass filtering the signal at $F_c$=0.5 Hz. Hence, the input signal for QRS detector has a range (0.5-100) Hz.

3.3.1. Moving Window Integrator Algorithm

The basic steps in the moving window integrator are differentiator, squaring algorithm, and summations. The block diagram of this calculation is shown in Fig. 3.5. The ECG output from the band-pass filter is differentiated to provide the QRS-complex slope information. Here, a 5-point differentiator with the transfer function is used.

$$H(z)=(1/(8T))*\{(-z^{-2}-2z^{-1}+2z^{1}+z^{2})\} \quad (3.1)$$

The difference equation is

$$y(n)=(1/(8T))*\{-x(n-2)-2*x(n-1)+2*x(n+1)+x(n+2)\} \quad (3.2)$$

Where, x(n) is the filtered ECG signal. The next step in the pipeline is the squaring function, where the above differentiator output signal, $y(n)$, is squared point by point. The operation is presented in equation (3.3).

$$z(n)=[y(n)]^2 \quad (3.3)$$

This squaring process makes all data points positive and does non-linear amplification of the output of the derivative emphasizing higher frequencies. The squaring process intensifies the slope of the frequency response of the derivative and helps to restrict false positives caused by T waves with higher than usual energies. Next step is the moving window integration whose purpose is to obtain waveform feature information apart from the slope of the R wave. It is calculated as follows:

$$w(n)=(1/N)*\{z(n-(N-1))+z(n-(N-2))+\ldots\ldots\ldots\ldots+z(n)\} \quad (3.4)$$

where N is the number of samples in the width of the integration window and z(n) are the squares of filtered values.

The width of the window is determined empirically; taking into consideration that the QRS complex occupies about one-sixth of a second (≈150 ms). In the present work, time-domain fiducial point detection is implemented with a sampling frequency ($F_s$) of 250 Hz and other algorithms including feature calculations and frequency domain spectral estimations are performed after converting $F_s$ to 500 Hz. For a sampling rate of Fs=250 Hz, the window length of 40 samples is chosen. Moving window integrator output contains information about both the slope and the width of the QRS complex. The upward slope of the moving window integrator for each beat is found to correspond to the QRS complex [Pan and Tompkins, 1985].

Several amplitude and interval thresholds are applied in QRS detection algorithm for making a decision. They include mean baseline amplitude (after removing QRS complexes), mean amplitude, maximum and minimum amplitudes of ECG signal and moving window integrator output, and so on. The location of previous QRS complex is taken into account during the search for the present QRS complex; *i.e.*,

the algorithm skips a blanking period of 200 ms after detecting a QRS complex. This interval is decided based on the physiology of the pace maker and pulse propagation inside the heart.

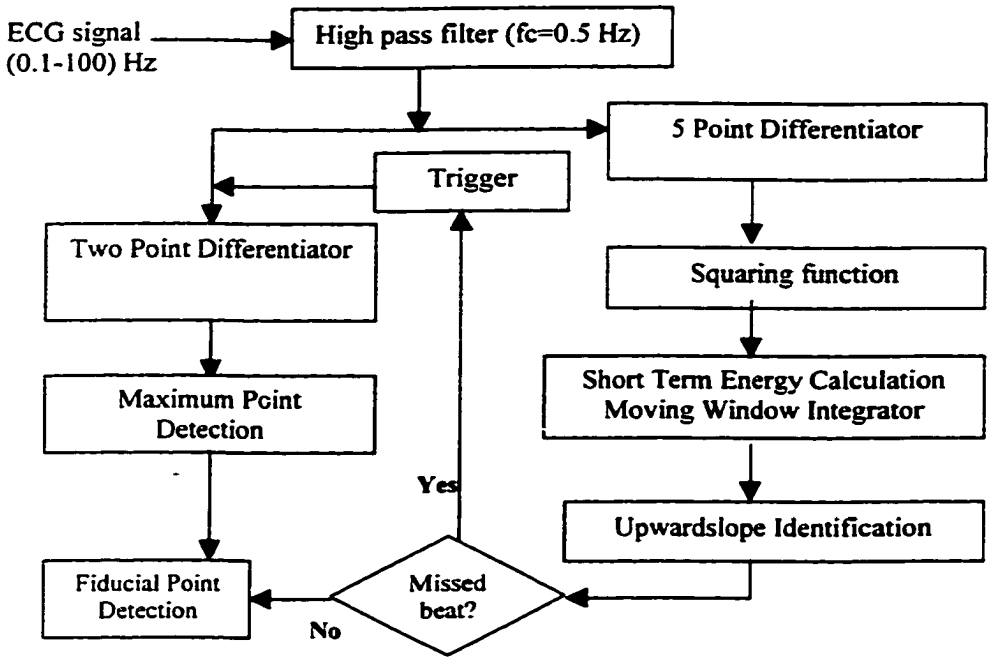



Fig. 3.5. Combinational algorithm for QRS detection

Due to the short duration of recordings, a learning phase of few seconds, as mentioned by Pan and Tompkins, 1985, is not available in present algorithm. Hence, the algorithm is not adjustable and is prone to minor errors. Two remedies are implemented in the present work:

(i) Backtracking during fiducial point detection to see if all the R points are detected properly.

(ii) Whenever a beat is missed (or) the algorithm does not find a beat for more than 1.5 seconds, control automatically switches to the second method of QRS detection, which makes use of maximizing the first difference.

3.3.2. Maximizing the First Difference

The moving window integrator tends to give poor performance for following cases;

(i) Beats dominated by noise.

(ii) A data segment with QRS complexes with varying morphologies.

(iii) Presence of very low amplitude QRS complexes in the middle of normal QRS complexes.

(iv) QRS complexes followed by very high T waves.

If one assumes a 'least informative' situation (*i.e.*, with uniform distributions), the estimation procedure carries out a nonlinear transformation of the linearly filtered ECG which after some simplifications, is given by

$$z(n) = \max_{\tau_1 \le T \le \tau_2} |y(n)-y(n-T)| \tag{3.5}$$

Where, $\tau_1$ and $\tau_2$ are positive integers. The signal $z(n)$ in equation (3.5) is thus the rectified, maximum amplitude difference between two samples.

Owing to the large computational requirements associated with equation (3.5), (the maximization is performed at each step in time n), an approximate scheme can be used in which $y(n)$ and $y(n-T)$ are chosen among the peaks and valleys in the filtered signal [Sornmo *et al.*, 1980]. Sometimes, the signals are suppressed rather than enhanced, as in split QRS complexes. Summing together the slope coefficients within a QRS complex eliminates baseline shifts; QRS complex yields a sum close to zero, but a baseline shift does not. According to Pan and Tompkins, [1985] and Sornmo *et al.*, [1980], the short-time energy produces a signal in which the noise background is better suppressed. In contrast, small amplitude QRS complexes tend to be too much

suppressed and hence, the need for the present method of maximizing the first difference. Complete results indicating the performance of QRS detection algorithms are presented in chapter 4 on results.

## 3.4. Overall Estimation of Fiducial Points

Fiducial points detected in the present work include P wave on- and off-set points, QRS complex on- and off-set points (Q start, Q point, R point, S point and J points) and T wave on- and off-set points. The detection is performed in three stages, namely

(i) Signal smoothing.

(ii) Wave detection.

(iii) Wave boundary identification for P and T waves.

### 3.4.1. Signal Smoothing

Input to the signal smoothing is the original ECG signal in the frequency range (0.1-100) Hz for MIT-BIH database and (0.05-100) Hz for clinical database. Signal smoothing is performed before detecting P and T waves and after R wave detection. A simple causal smoothing makes use of summation operation for the few previous points. Equation (3.6) exactly indicates such a causal relation.

$$y(n)=(x(n)+x(n-1)+\ldots\ldots+x(n-N))/N \qquad (3.6)$$

The smoothing operation is a low-pass filter operation and eliminates noisy and turbulent components. Detection of both peaks and edges of the component waves becomes easier after smoothing. An added advantage is the availability of a feature to indicate the noisy beats, calculated as the ratio of variance before and after smoothing.

The present smoothing filter implements a three-point smoothing. Noisy beats seem to have a higher range compared to normal beats.

3.4.2. Component Wave Detection

The algorithm below indicates step-by-step identification of fiducial points, starting from QRS detection. The algorithm makes use of some information from previous beats (after the first beat).

Step 1: Single lead QRS detection, using moving window integrator algorithm, is to be done. In the presence of large S or Q waves and dominant T waves, or when the QRS complex is missed for more than 1.5 seconds. A search is performed using the first-order differentiation and maximizing the output.

Step 2: Calculate tentative boundaries of QRS complex using bottom and top of the rising arm of the moving window integrator output and search for R point. In case of maximizing the first difference, approximate the boundaries. If QRS characteristics differ, go to fiducial point calculation for ventricular beats, *i.e.*, alternate steps 3a - 6a.

Fig. 3.6.(a) and 3.6.(b), shows the location of R-point and QRS complex boundaries in ECG signal, with respect to the rising arm of the moving window integrator output. Fig.3.6.(a), illustrates for a single beat and Fig. 3.6.(b), for a sequence of beats. Ability of the algorithm to adjust for ventricular beats is illustrated in Fig. 3.6.(c). A sample case showing a difficult ECG segment in using moving window integrator algorithm is illustrated in Fig. 3.6.(d). Plots are drawn for magnitude in Y-axis versus number of sampling points in X-axis. Magnitudes are based on analog-to-digital converter discrete values, and they can be converted to any other scale easily.

The location of the peak inside a QRS interval indicates R point in moving window integrator algorithm and the location of the peak inside an interval in the neighborhood of the location of the maximized first difference. Hence, a comparison of the outputs of both the algorithms are presented in Figures 3.6. and 3.7.

Fig. 3.7. shows samples of detector outputs making use of the "first-difference maximization" principle. The diagrams illustrate the relation between the original signal and their maximized difference signals. The maximum values in the first-differences of ECG signal are used to detect the approximate location of R points. This algorithm takes over when there is a problem of thresholding as well as in situations mentioned in section 3.3.2.

Step 3: Search along the right side of the detected R point for S point, based on duration, zero-crossing of slope and angle between the lines formed by successive points. The formula for angle between two lines, given their start and end points is provided by $\theta(i) = atan((m1+m2)/(1-m1*m2))*(180/pi)$ (3.7)

where, $m1 = dat(i)-dat(i-1)$ and $m2=dat(i+1)-dat(i)$

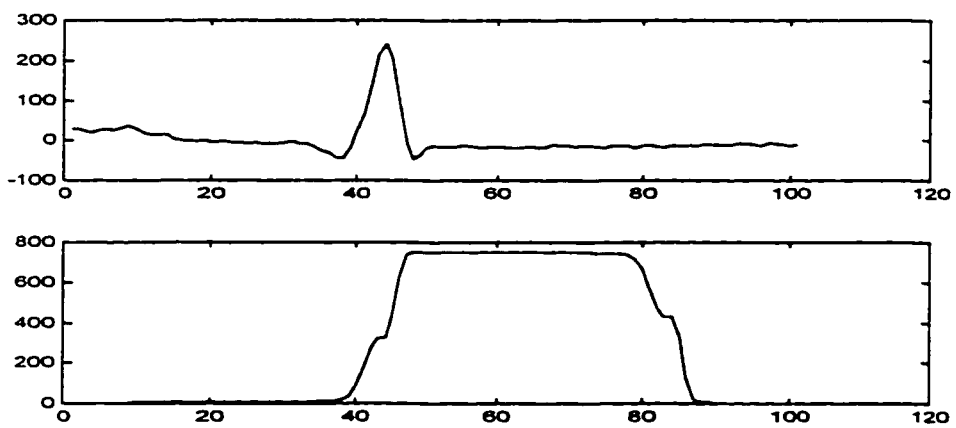



Fig. 3.6.(a). Normal ECG beat and relative position of QRS complex with respect to moving window integrator (**mwi**) output

Fig.3.6. Moving window integrator output for ECG signals

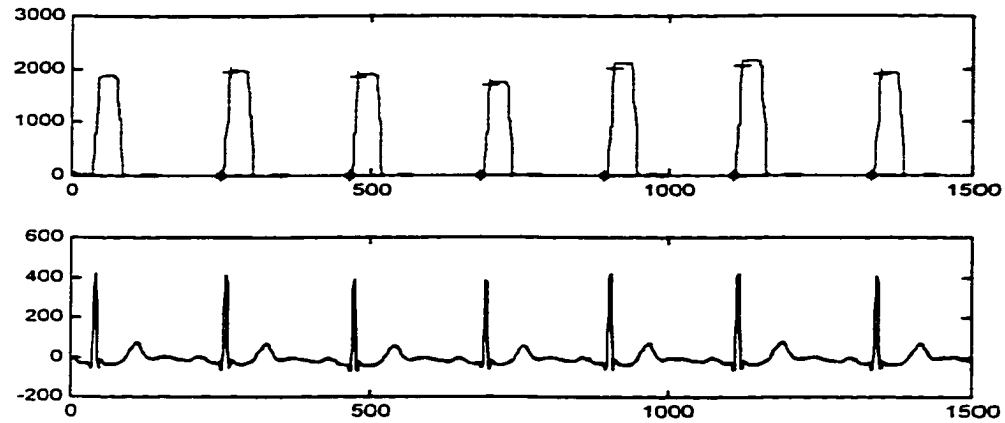



Fig. 3.6.(b). A sample data segment with sequence of normal beats and its mwi output

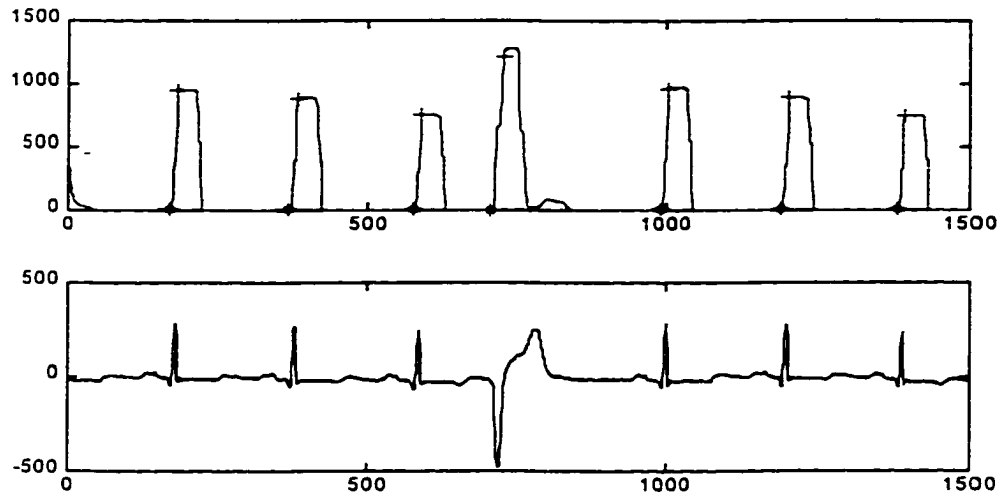



Fig. 3.6.(c). Ability of mwi algorithm to accommodate PVC beats.

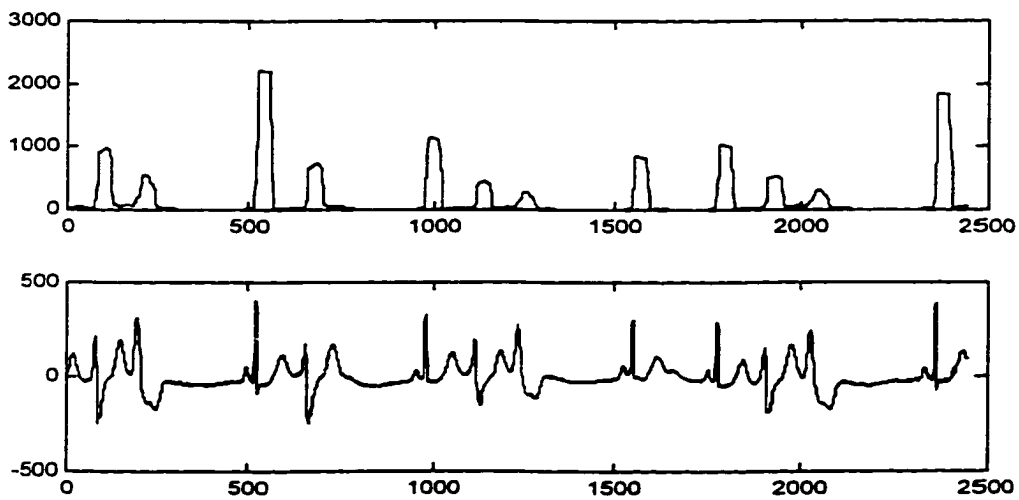



Fig. 3.6.(d). Difficult cases for mwi algorithm in deciding a proper threshold
Fig.3.6. Moving window integrator output for ECG signals (Cont.)

Step 3a: Presence of premature ventricular beat is assumed. Large negative Q wave (Qr) is detected from the tentative QRS boundary calculations.

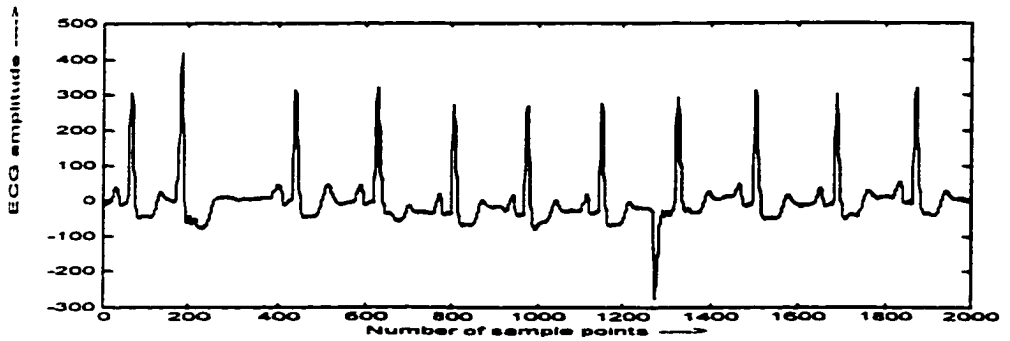



Fig. 3.7.(a). A sample ECG segment - m1051

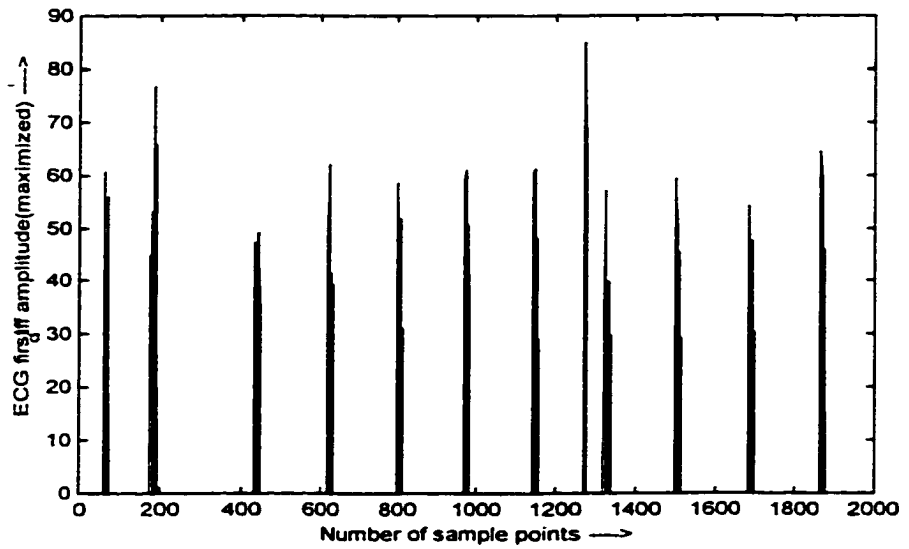



Fig. 3.7.(b) Maximized first difference of the ECG signal - m1051

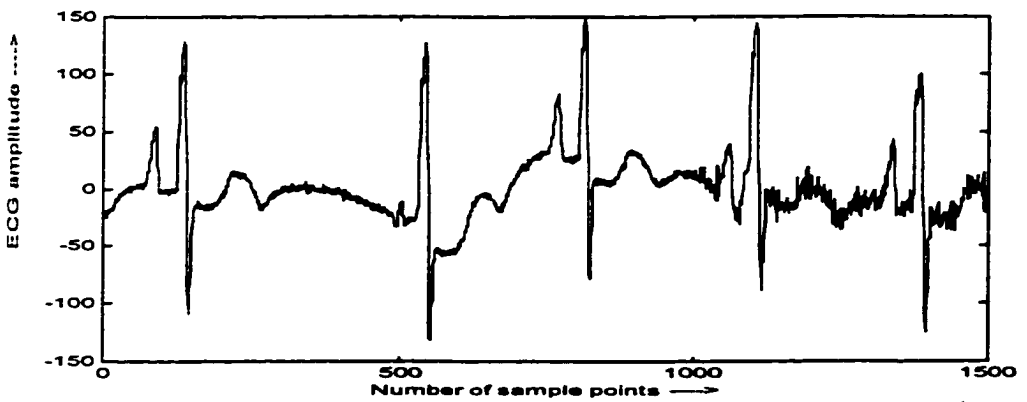



Fig. 3.7.(c). ECG segment with noise, with base-line drift - m1084

Fig.3.7. Output of maximized first difference for ECG signals

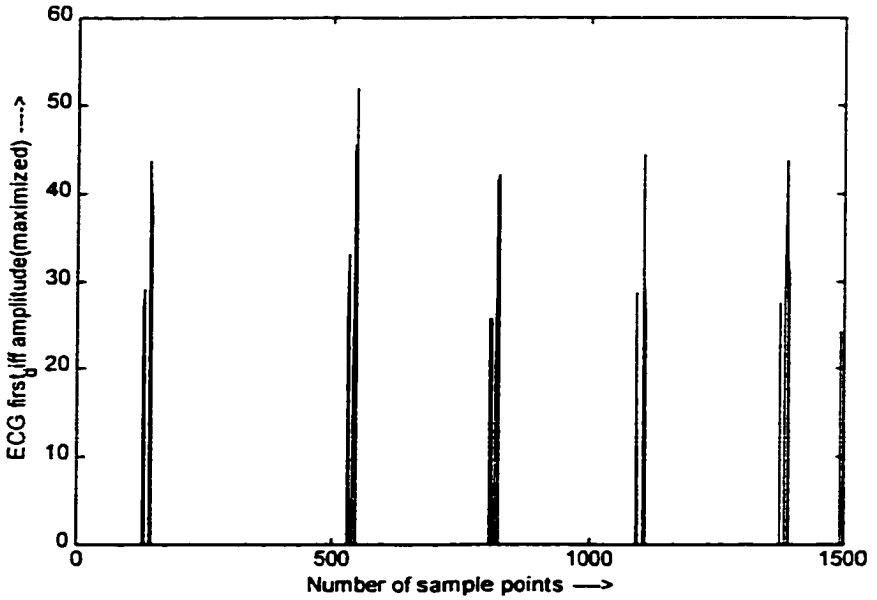



Fig. 3.7.(d). First_difference (maximized) of ECG signal - m1084
Fig.3.7. Output of maximized first difference for ECG signals (Cont.)

Step 4: The J point is defined as the junction between a QRS complex and its ST segment. The detection of a J point is also based on the angle and direction of the slope change. The changes are small compared to S wave detection.

Step 4a: The S point is chosen as the point of baseline crossing. The J point location varies with the location of S point.

Step 5: Search for T wave peak based on the definition of peak, after smoothing of the curve. A smooth peak shows a maximum with at least two descending points on either side as well as a sign change in slope at the peak. Another factor is that amplitude of the peak should be above an threshold amplitude, which is adjustable, based on R amplitude. Local baseline and the amplitude difference with respect to baseline are used and hence, even inverted T waves are detected easily.

Step 5a: T wave is detected using simple peak calculation after the J point and before the next beat.

Step 6: Proceed with the P wave detection and the process is similar to T wave peak detection, except that the search is in the other direction of QRS complex. A peak is defined as a local maximum where the sign of the derivative changes and smooth descent occurs on either side. With this step, the first stage of wave detection gets completed for the normal beat. The absence of a P or T peak is also noted. Crosschecks are introduced for the negative wave detection.

Step 6a: If the P wave is absent, the P wave start and end points correspond to the same point. The points are positioned after previous T point and before the start of present beat (Q start point).

### 3.4.3. Wave Boundary Identification

The previous step detects P wave and T wave peaks alone and also detects if those components are present or absent. The wave boundary identification step, on the other hand, detects boundaries of the waveforms, relative to the peaks. An important finding of the CSE study is that the algorithms tend to locate the end points of the T wave significantly earlier than human experts do [Willems *et al.*, 1990]. The end points of the P or T waves are defined in terms of average baseline amplitude in the present work. Baseline amplitude is approximated as the mean value after removing the QRS complex and the value tends to be closer to actual baseline. The end points (on- and off-sets) tend to merge smoothly with the baseline, and this fact is made use of in the present detection of end points. A derivative value of around zero occurs nearer to end points, compared with the rising and falling arms of P and T waves. Sometimes, a derivative sign change occurs, but it is not a reliable indicator. Hence, a combination of

criteria based on curvature, slope and merging with baseline are used for endpoint detection for both P and T waves.

In cases where a T wave of the previous beat merges with the P wave of the present beat, it is essential to distinguish the P and T waves in order to clearly detect the start of a beat. In such cases, detection of T wave end point provides the reference for finding P wave start point. It becomes essential to find such characteristic points even in the absence of P waves. A fiducial point is assumed for cases like premature ventricular contraction (PVC) that have no visible P waves. Inversion of waveforms causes another problem. Inversion of P and T waves usually do not matter with respect to baseline amplitude and end points usually return back to baseline level. However, waveform asymmetry in the component waves causes problems for algorithms, especially the asymmetry between ascending and descending arms of the waves. A few typical cases of detected fiducial points are shown in next chapter on results. Exact locations of the fiducial points are needed for calculation of both time and frequency domain features. A list of Matlab® algorithms developed in current research and a brief description of input and output relations in those algorithms are provided in Appendix C. The entire implementation is carried out in this Matlab® environment.

## 3.5. ECG Features Calculated in Time Domain

Time-domain features calculated in the present work are listed below along with their definitions:

(1) **PP interval (T1)**: Provides an indication of beat duration. Most other features are calculated inside this interval. For a segment with 'n' detected beats, (n-1) PP

intervals are present and other features are calculated for these (n-1) beats. The last beat is discarded, except for the start of P wave, i.e., $P_1$ point.

(2) **Mean base-line (T2)**: Mean base line is approximated as the mean amplitude of the whole beat, excluding the QRS complex. This value is a simple estimation and closely approximates the true base line of the beat.

(3) **QRS shape and sign (T3)**: This feature provides an indication of the possibility of ventricular beats. Three different shapes are distinguished: an upright QRS complex, an inverted QRS complex and one in between with large S wave.

(4) **Ratio of areas under QRS and T waves (T4)**: Calculation is shown in equation (3.8.) below.

$$R = \frac{\text{area under QRS complex w.r.to baseline}}{\text{area under T wave w.r.to baseline}} = \frac{\sum_{i=Qs}^{J} (qrs(i) - baseline)}{\sum_{i=T1}^{T2} (t(i) - baseline)} \qquad (3.8)$$

This parameter shows variation between beats with normal upright QRS and T waves and beats showing inverted waves.

(5) **JT1 interval (T5)**: This is a small duration separating QRS and T waves. Exact detection of J point complicates the estimation of this interval [Wagner.G., 1994]. The J point is the point of intersection between QRS complex and ST segment [Greenhut.S.E. *et al.*, 1989].

(6) **ST segment slope (T6)**: An iso-electric ST segment is present in a normal ECG beat. But, under diseased conditions like myocardial infarction (MI) or during exercise periods, the ST segment is displaced and presents varying shapes like concave, convex, oblique and plateau.

(7) **PR interval (T7)**: Standard feature indicating the approximate time of conduction between atrial and ventricular activations.

(8) **T wave inverted or upright (T8)**: A value of 0 or 1 indicates the regular or inverted T wave. Calculation is based on local base-line amplitude.

(9) **TP interval (T9)**: Standard feature indicating the duration between the end of ventricular activity and the start of next atrial depolarization. This parameter provides an indication for premature beats, bradycardia, and tachycardia.

(10) **P wave duration (T10)**: Calculated between the end-points ($P_1$-$P_2$) of P wave. This interval provides an indication of the premature beats, as well as presence or absence of P waves.

(11) **Ratio of upward versus downward slope for QRS complex (T11)**: Another feature about shape and duration of QRS complexes, this feature especially useful in identifying RBBB and LBBB classes.

(12) **Ratio of R amplitude and Q amplitudes (T12)**: The amplitudes are calculated with respect to base-line amplitudes. This ratio enhances the information provided by feature (T11) and feature (T3).

(13) **QT interval (T13)**: This interval is the time between Qs and $T_2$ points and it provides an approximate duration of ventricular activity. Bonadonna, 1998, has performed an extensive study on the variation of QT interval with respect to other intervals.

(14) **Noise reduction after smoothing (NRS) (T14)**: The effect of smoothing in reduction of variance seems to be more in the case of noisy beats compared to non-noisy beats. This parameter is calculated as in equation (3.9) below.

$$NRS = \frac{\text{Variance of the original signal}}{\text{Variance of the signal after double smoothing}} \tag{3.9}$$

(15) **Number of zero-crossings (T15)**: Provides an easy quantifier for atrial flutter and fibrillation activities and is calculated with respect to local base line.

(16) **Previous TP interval (T16)**: This parameter is essential to detect the occurrence of premature beats. This is the only feature that extracts the information beyond the present PP interval.

The next chapter on results provides typical ranges of values for all the above features and indicates the abnormal cases of feature values which lead to discarding particular beats from further analysis.

## 3.6. ECG Features in the Frequency Domain

Attempts in frequency domain feature extraction of ECG signal are fewer, compared with time domain implementations. Recent interest focuses on eliminating or detecting particular waveforms corresponding to atrial or ventricular activities [Langberg *et al.*, 1998]. Other areas of interest in frequency domain include high-frequency late potential analysis and identifying ventricular tachycardia, flutter and fibrillation [Steinbegler *et al.*, 1989; Minami *et al.*, 1999]. A major problem in traditional frequency domain analysis is the inability to localize the rhythm disturbances in individual component waves in an ECG beat. The dominance of QRS spectral power over the other segments in an ECG beat adds to the problem [Murthy and Niranjan, 1992]. However, the increasing ease and speed of computation using high-speed personal computers make the frequency domain parameters rather easy to calculate.

The resolution of parametric estimator seems to be better compared to periodogram estimator, especially for short data length [Marple, 1987]. The mathematical relation and comparison between the two estimators has been measured by the minimum frequency separation of two equi-amplitude sinusoids embedded in white noise for which the two sinusoids can be discerned. A comparison of resolution of both estimators is shown in Fig. 3.8., with respect to data record length and signal-to-noise ratio. An autoregressive spectral estimator shows a better performance for shorter data length [Kay, 1992].

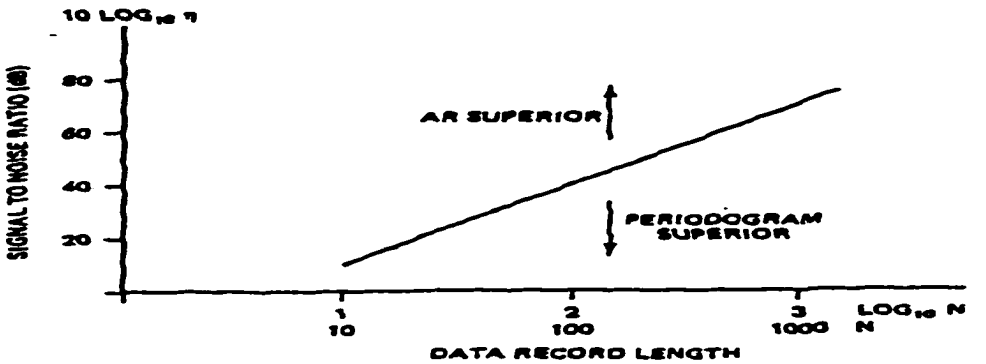



Fig. 3.8. Comparison of resolution of periodogram and AR estimators with respect to data record length and signal to noise ratio [Kay, 1987].

3.6.1. Model-based Combination Approach

Power spectral estimation in the current scheme uses a combination of modeling and subtraction of individual components. Recent techniques perform beat-by-beat analyses in the frequency domain using cancellation techniques [Langberg *et al.*, 1998; Shkurovich *et al.*, 1998].

In this work, six different types of spectral estimations are performed. The first estimation is based on original beat and no subtraction of individual components is

performed. Burg's algorithm is used to estimate the power spectral density with order P=32. The Burg's algorithm estimates the reflection coefficients and then uses the Levinson recursion to obtain the AR parameter estimates. If estimates of the reflection coefficients $\{k_1, k_2, k_3, \ldots\ldots k_p\}$ are available, the AR parameters may be estimated as shown in the block diagram in Fig. 3.9.

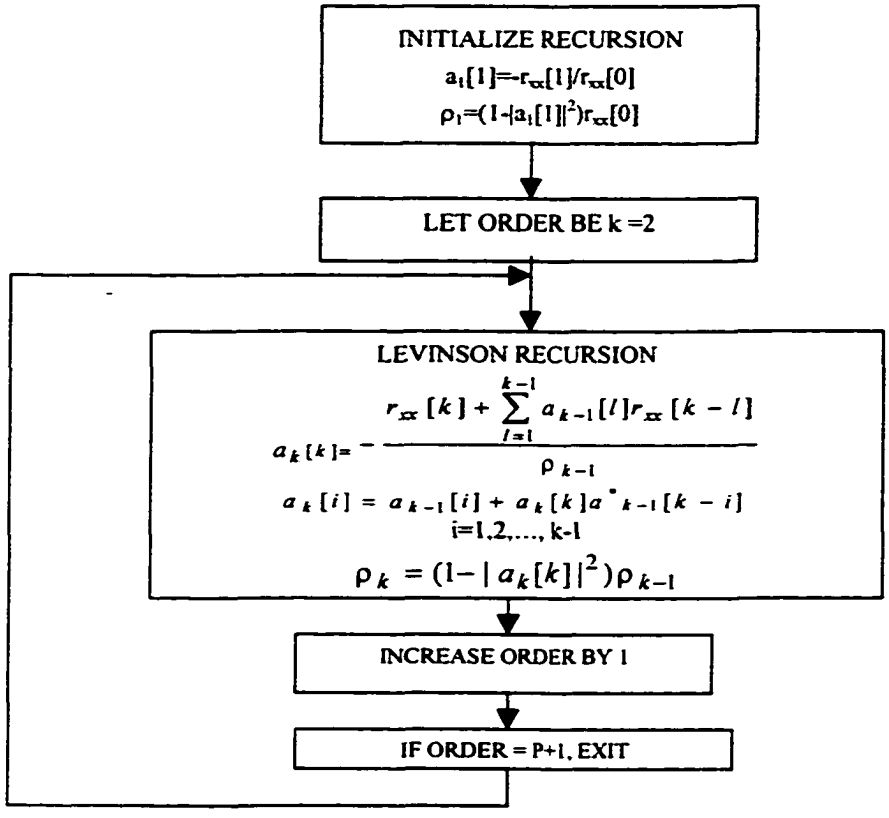



Fig. 3.9. Estimation of AR parameters in Burg's algorithm

A combination approach involving simulation and subtraction of individual components in a beat is applied to generate five other spectral estimations. Each beat is modeled as a linear combination of the normal sinus rhythm component and a rhythm disturbance component.

$$beat(t)=nsr(t)+disturb(t) \tag{3.7}$$

Where, beat(t) = an observed beat

*nsr(t)* = normal sinus rhythm beat of matching timings and

*disturb(t)* = disturbances in the beat

The normal sinus rhythm component is simulated based on amplitude and interval calculations of each subject's observed ECG beat. A normal sinus rhythm beat corresponding to an observed PP interval is simulated using Matlab®-Simulink algorithms. A linear superposition of P waves, QRS complex and T waves with standard timings and with amplitude matching are used to simulate waveforms. Standard variations in time intervals like QT interval, PR interval and TP intervals are derived from literature [Bonadonna, 1998; Wagner, 1994].

A block diagram representation of the simulation is shown in the Fig. 3.10. The simulation model recognizes that major interval changes occur more in base-line intervals compared to wave duration. Hence, changes in heart rate and PP intervals impact the TP interval most strongly, followed by PR interval and ST segments, in that order.

The resulting difference, *disturb(t)*, is modeled using an auto-regressive (AR) model using Burg's algorithm. An autoregressive model of order P=32 is used for modeling and spectral estimation. The number of points chosen for the estimation is 1024 in order to provide adequate resolution. The choice of the order is based on

(i) Ability to repeat the shape of frequency spectrum based on more than 150 beats, as estimated by Thakor *et al.*, 1984. A model order of P=32 seems to provide the same accuracy as the averaged spectrum estimated by the above group.

(ii) Stability in estimation of the spectrum. Any increase in order beyond P=32 produces negligible advantage.

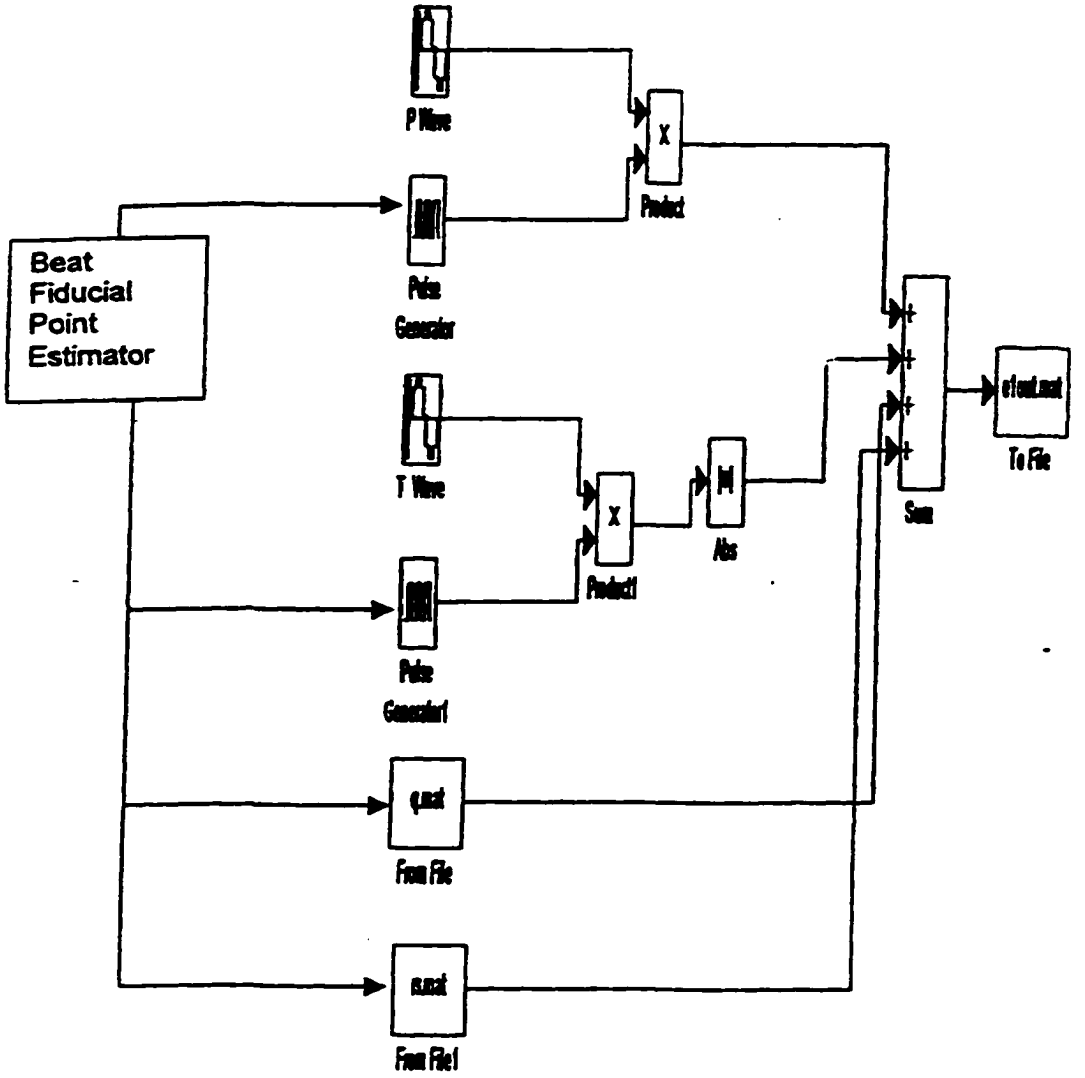



Fig. 3.10. Simulation of normal ECG beat from calculated values of fiducial points

In order to localize spectral information in individual components, the above spectral estimation is repeated for the same beat, after the removal of individual components like P wave, T wave and QRS complexes. Final spectral estimation involves the removal of all three components for estimating noise, artifacts and/or any

other major rhythm problems. Hence in all the following six spectral estimations are performed;

(i) Spectrum of original beat

(ii) Spectrum of (original beat - simulated normal beat)

(iii) Spectrum of (original beat - P wave)

(iv) Spectrum of (original beat - QRS complex)

(v) Spectrum of (original beat - T wave) and

(vi) Spectrum of (original beat - (P+QRS+T) waves)

Each estimation, twelve features are estimated and they include mean power frequencies, max power frequencies and power spectral density ratios in the overall spectral band as well as for smaller bands. The result is a collection of 72 feature values. A simple feature selection procedure to reduce the number of useful features is discussed next.

3.6.2. Feature Selection in the Frequency Domain

A test data set of 350 beats including 10 simulated beats (one for each class) is chosen for selecting features. The annotated beats belong to normal (NOR), premature ventricular contraction (PVC), atrial premature contraction (APC), right bundle branch block (RBBB), left bundle branch block (LBBB), inverted T (INVT), ST variations (ST), paced beats (PACE), noisy beats (NOI) and other beats (OTH). Based on visual criteria on the spectrum as well as the ability to distinguish between at least three classes, a simple expert system is developed with VP-Expert® system shell. Visual criteria include the ability to highlight the disturbance component and produce large spectral amplitude. Major classification information is found to be provided by spectral

estimations (i) and (ii) [Srikanth *et al.*, 1999]. Results of the procedure are presented in next chapter along with sample spectral plots. Another reason for eliminating spectral estimations (iii), (iv), (v), and (vi) is that they are prone to errors in detection of fiducial points as well as non-stationarity constraints.

3.6.3. Selected Features in Frequency Domain

Spectral estimations (i) and (ii) alone are taken for further calculation of features. Selected frequency domain features include

(1) **Mean power frequency of the original beat(mpf1) (F1)**: This feature indicates the distribution of power across the frequency band of (0-125) Hz. The formula for calculating mpf1 is given in equation (3.8).

$$mpf1 = \frac{\sum_{f(i)=0}^{f(i)=125} P(f(i)).*f(i)}{\sum_{f(i)=0}^{f(i)=125} P(f(i))} \tag{3.8}$$

Where, $P(f(i))$ indicates the power spectral densities at frequencies f(i).

(2) **PSD ratio for the band (0.5-6) Hz for the original beat(psd1) (F2):** This is the band where most of the power is concentrated in original beat [Thakor *et al.*, 1984]. This parameter shows variations for ventricular beats (reduction), atrial flutter and atrial fibrillation.

(3) **PSD ratio for the band (6-12) Hz for the original beat(psd2) (F3):** This band shows an increase in the power content for classes like ventricular tachycardia, flutter and fibrillation.

(4) **PSD ratio for the band (12-18) Hz for the original beat(psd3) (F4):** Most of the spectral power is contained with in this limit of 18 Hz for normal beats. The rest of the

power is negligible usually for classes chosen in present work. The relations between the above three parameters *psd1*, *psd2*, and *psd3* are given below.

$$psd1+psd2+psd3+rest=1 \tag{3.9}$$

where, *rest* indicates the power contained in rest of the frequency band (18-250) Hz.

(5) **Mean power frequency of the (original-simulated) beat(mpf2) (F5):** Indicates the shift in mean power frequency after removing the simulated beat. Both increase and decreases are possible from the original *mpf1* calculation, depending on the beat class.

(6) **Mean power frequency of (0.5-10) Hz band(mpf3) (F6)**

(7) **Mean power frequency of (10-20) Hz band(mpf4) (F7)**

(8) **Mean power frequency of (20-30) Hz band(mpf5) (F8)**

All three mean power frequency estimations mpf3, mpf4, and mpf5 show considerable sensitivity for different classes.

(9) **PSD ratio for (0.5-6) Hz band for (original beat-simulated beat)(psd4) (F9)**

(10) **PSD ratio for (6-12) Hz band for (original beat-simulated beat)(psd5) (F10)**

(11) **PSD ratio for (12-18) Hz band for (original beat-simulated beat)(psd6) (F11)**

Parameters F9, F10 and F11 provide information about the impact of QRS complex relative to rest of the beat. The information is amplified by the subtraction of a simulated beat from the original beat. These bands were found to be useful for detecting ventricular tachyarrhythmia by Minami *et al.*, 1999. They also provide useful information for atrial flutter and atrial fibrillation beats.

The normal ranges of values for all the above features are presented in next chapter and Appendix-C provides information on programs developed for the feature

extraction process. In each beat, a total of 27 features are calculated, 16 in time domain and 11 in frequency domain.

### 3.6.4. Overall Input Feature Set

Input features are organized as a matrix of the order [M×N] where, M=number of beats and N=number of features/beat. For each beat, the organization of the features and organizational details are as follows:

$$F=[Feature(t);\ Feature(f);\ Data\ index;\ Beat\text{-}index;\ Group\ index;\ Sub\text{-}class\ index] \quad (3.10)$$

$$F=[1\times16;\ 1\times11;\ 1\times1;\ 1\times1;\ 1\times1;\ 1\times4] \quad (3.11)$$

Where, *Feature(t)* = time-domain feature set and

*Feature(f)* = frequency-domain feature set.

The overall number of columns thus equals 34. Appendix-B provides a sample of organization of data including annotated beat types and codes. The entire set of features may not be useful for classifying all beat classes chosen, and only a sub-set is chosen for individual group of classes.

## 3.7. Data Set and Feature Set

Electrocardiogram data used in the present work are mostly derived from standard MIT-BIH database and routine clinical ECG data recorded from Kilpauk Medical College Hospital, Chennai, India; and Madras Medical College Hospital, Chennai, India. All chosen data are derived from limb lead II or modified limb lead II. Apart from these data, a small collection of ECG data with different morphological variations was simulated using the Matlab-Simulink® package. At least two independent physicians and/or cardiologists annotated each beat in all the data segments

chosen for the study. The simulated data are used in formative evaluation and in system development.

3.7.1. About MIT-BIH data set

The MIT-BIH ECG database has been accepted as a standard database for developing and testing ECG processing and pattern recognition algorithms. This database contains 48 records, each of which is slightly over 30 minutes duration. Two groups of data are in the MIT-BIH database. The first group, numbered 100-124 inclusive with some numbers missing, serves as a representative sample of the variety of waveforms and artifacts that an arrhythmia detector might encounter in routine clinical use.

ECG data in the second group, numbered from 200-234 inclusive (again with some numbers missing), were chosen to include complex ventricular, junctional and supraventricular arrhythmia and conduction abnormalities. Several of these records are selected because of the features of the rhythm QRS morphology variations, or because of their signal quality, which may be expected to present significant difficulty to arrhythmia detectors [Moody, 1992].

3.7.2. Selection of the Lead

Nearly 46 out of 48 data records in the MIT-BIH database include the data from limb lead II. Limb lead II is available in all 3-lead as well as 12-lead routine ECG equipment. Further, prominent and upright QRS complexes facilitate the choice of this lead. A brief survey of top ten commercial routine ECG monitors also indicate the

dominant presence of limb leads, and especially lead II positions [HME *Inc.*, 1999; Jenkins' ECG library, 1998, Marquette *Inc.*, 1999].

Schijivenaars *et al.*, 1996, assessed stability of automated ECG interpretation with respect to high susceptibility in chest electrode positions. A high susceptibility to chest electrode positions may cause a program to generate entirely different interpretations on ECGs differing slightly in electrode positions. Hence, for routine ECG recordings limb electrodes seem to be the preferred positions. The present work uses the limb lead-II as the lead for the single-lead study.

### 3.7.3. Data set – Statistics

Table.3.1. provides the details regarding the final data set, including both MIT-BIH and clinical data, chosen for analysis. One hundred twenty beats are added for VES, WPW, JUN and AFL classes after the first testing due to lack of generalization. Reasons for the addition of these beats are explained in section 3.8.3.

### 3.7.4. Input Feature Set

The ultimate objective of feature extraction is to obtain a pattern space consistent with low-dimensionality, retention of sufficient information, and enhancement of distance between classes in pattern space. A possible way to evaluate the usefulness of a feature space is to employ a comprehensive ECG database. A complement to this approach is to use a mathematical model for simulation of ECG data. In the present work, both approaches are followed. Features in time domain are mostly used for pattern recognition but frequency domain features are added on a need-only basis.

Table. 3.1. Statistics on data set

| Parameter | Numerical value |
|---|---|
| Total number of data segments | 495 |
| Total number of beats | 3734+120* |
| Selected beats | 3669+120* |
| Percentage discarded | 1.75 |
| Normal (NOR) beats | 1065 |
| Right bundle branch block (RBBB) beats | 220 |
| Left bundle branch block (LBBB) beats | 211 |
| Ventricular escape (VES) beats | 27+40* |
| Premature ventricular contraction (PVC) beats | 428 |
| Paced (PACE) beats | 106 |
| Atrial premature contraction (APC) beats | 199 |
| ST/T abnormalities (ST/T) beats | 600 |
| Wolf-Parkinson-White (WPW) (Pre-excitation) beats | 30+20* |
| Junctional (JUN) beats | 64+30* |
| Atrial fibrillation (AFIB) beats | 388 |
| Atrial flutter (AFL) beats | 89+20* |
| Noisy/base-line wander (NOI) beats | 242 |

*Number of extra beats added, to ensure adequate representation

## 3.8. Neural Network Implementation

Multi-layer feedforward neural networks are the most used networks for pattern recognition tasks. Hornik, 1991 showed that with continuous, bounded and non-constant activation functions, the continuous mappings can be learned uniformly over compact input sets. Similarly, a linear associator introduced by J. Anderson and Kohonen independently [Hagan *et al.*, 1995], can associate or learn Q pairs of prototype input/output vectors:

$$\{p_1,t_1\},\{p_2,t_2\},.....,\{p_Q,t_Q\} \qquad (3.21)$$

Multilayer feedforward networks are otherwise called multilayer perceptrons. In the present work, multilayer feed-forward networks are used for classifying an ECG beat whereas linear associators are designed for relating the classified beats to the closest rhythm statements. Linear associators perform with much accuracy, when the number of rhythm classes is known. Classification of rhythms in present work is a deterministic process, when the underlying beats are identified correctly. Such tasks do not need the complexity of feed-forward networks.

### 3.8.1. Overall Classification Problem

Multi-layer feed-forward networks provide the simplest implementation among the family of neural networks. The problem is designed in such a way that an optimal combination of smaller feed-forward networks may work on groups of smaller set of similar beats. Two options are possible to solve the classification problem. The first option involves using a single feed-forward neural network, which should be able to classify the chosen thirteen ECG classes, as listed in Table.3.1. However, this specification leads to a large neural network with more hidden neurons and internal connections proportional to multiples of number of inputs. The major problem in using a single feed-forward network seems to be its inability to adjust to wide variations in feature values [Haykin, 1999]. Another peculiar problem in ECG feature classification is the accuracy of features and their relation to exact location of fiducial points. Exact location of fiducial points is difficult in many of the beats belonging to atrial fibrillation, flutter, ventricular beats and in noisy beats. Hence, an accurate set of time-domain features is not available for all thirteen classes.

A strategy has to be employed where time-domain features are used for classifying mostly supraventricular arrhythmia, and a combination of time and frequency domain features are used for other classes including noisy beats. Such an arrangement of dividing the problem into smaller problems helps to utilize the inherent ability of feed-forward networks to learn the minor variations of feature values and generalize. Based on the above considerations, the problem is divided into four sub-problems, which means the data set also has to be divided into four sets.

3.8.2. Sub-Problems and Smaller Feature Sets

The whole set of classes is divided into four groups as shown below in Fig. 3.11. Exact definitions and a few sample segments are presented in Appendix-A. Four different sets of features are chosen for identifying each of three/four sub-classes among each group. The indexes for features are as indicated in earlier sections on time and frequency domain features in sections 3.5. and 3.6.3.

The initial sets of features had eleven features for each of four groups, selected according to famous Minnesota Coding by Blackburn, 1999 and Kors *et al.*, 1996. Novocode by Rautaharju *et al.*, 1998 as well as "Marriot's Practical Electrocardiography" by Wagner, 1994, are also used for the selection of appropriate features. Frequency domain features are selected based on Minami *et al.*, 1999, Strohmenger *et al.*, [1997], and Schkurovich *et al.*, [1998]. The final feature sets used to classify with in each group are shown in Table. 3.2.

3.8.3. Practical considerations and strategy

Selection of a final set of features, number of hidden neurons, and training goal in mean square error, *etc.*, has always been an art in neural network implementations. An attempt is made to provide a working logical schematic for the present neural network implementation. Some useful strategies included the selection of features, which show sufficient variation between three or four sub-classes inside each group (using t-test). Another approach followed throughout is the use of scatter plots provided in Matlab®. A sample diagram for group 1 is shown below in Fig. 3.12., showing relations between a particular feature variation with respect to the three sub-classes in group 1.

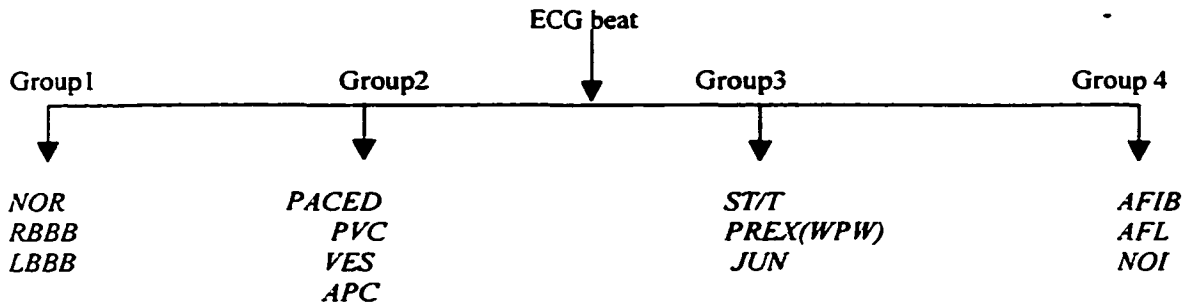


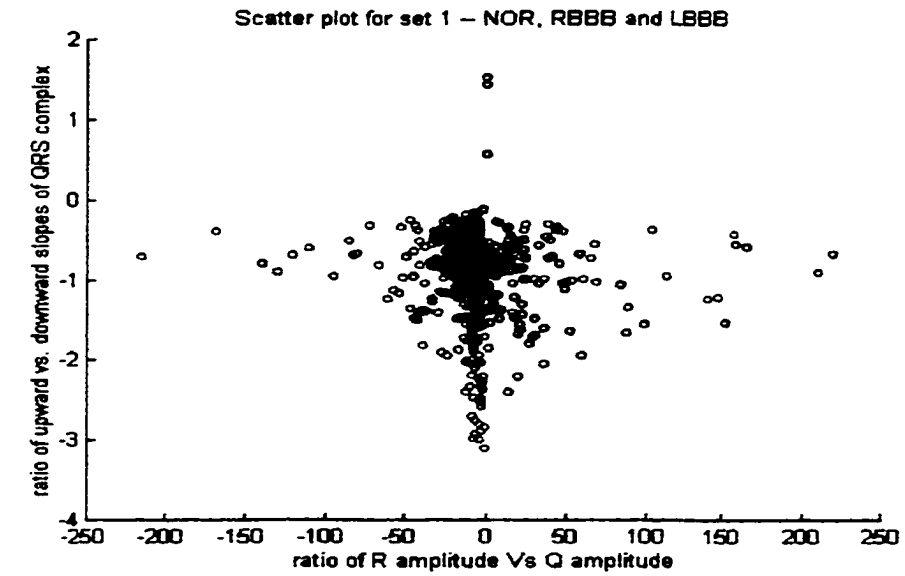

Fig. 3.11. Division of data into four major classes with their sub-classes

Table. 3.2. Set of features for classification inside each group

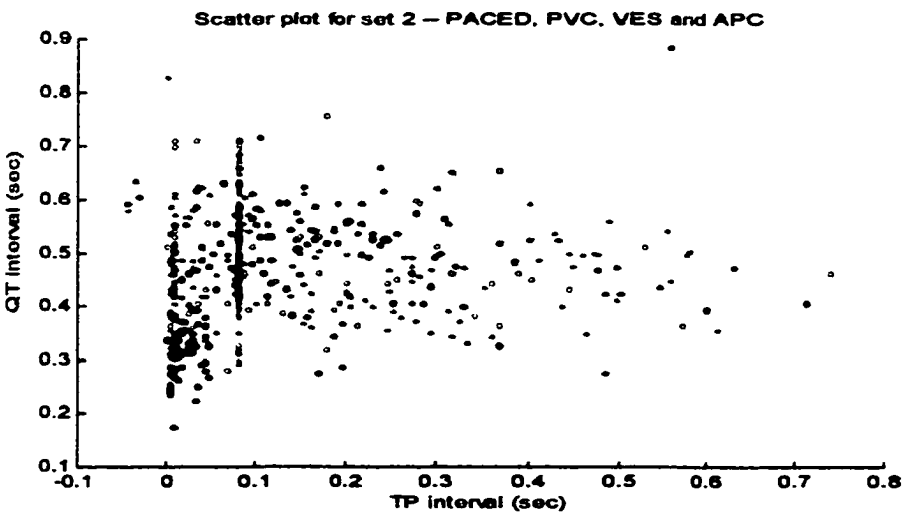
| | |
|---|---|
| *Group 1* | *[T3;T4;T5;T7;T11;T12;T13]* |
| *Group 2* | *[T9;T13;T16;F1;F2;F3;F4]* |
| *Group 3* | *[T5;T6;T7;T8;T9;T13;T16]* |
| *Group 4* | *[T1;T10;T14;T15;F1;F5;F6;F7]* |

The final sets of features for each group are based on the complete permutation of eleven features in smaller sets of six or seven or eight features. Some essential

features are fixed, but others are replaced each time. A new neural network with particular input features is implemented. Performance of the network in training is monitored. The sets of features providing better performance in training are chosen. Inputs to the final networks contain the least number of features.

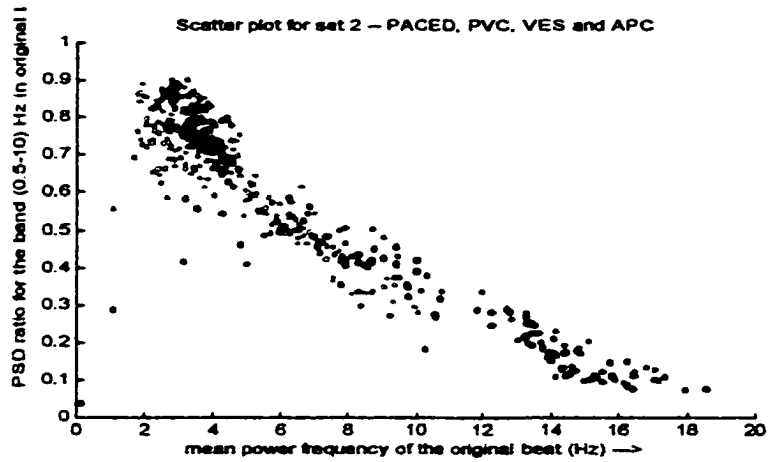



(a)




(b)

Fig. 3.12. Scatter plots of the distribution of features with respect to classes




(c)

Fig. 3.12. Scatter plots of the distribution of features with respect to classes (Cont.)

Selection of number of hidden neurons for a particular task is another topic where practical experience comes in handy in the absence of established theory. An ideal value is to choose the number of hidden neurons such that they satisfy the following two competing standards;

(i) A very low mean square error (MSE) in training and

(ii) A very high generalization ability for new set of test data.

A similar approach is followed in statistical discriminant analysis [Dallas.E.J., 1998]. Too low a value of MSE indicates an over-fit, and trouble arises when unknown data are chosen for testing. If the number of hidden neurons is high, the fit occurs easily for training data and testing performance decreases. Hence, an optimum mean square error target for training seems to be in the range 0.10-0.20. Only a few inputs present large errors and most of the inputs to the networks reach their target output values. The network weights are sufficiently flexible and the testing performance is usually better.

A large set of results, checking different numbers of hidden neurons as well as combinations of features, is provided in Appendix-E. The neural network providing the lowest mean square error is chosen as the final choice. The chart below provides a precise step-by-step approach followed in present work to design a feed-forward neural network.

Step 1: Pre-processing the input set of features (Normalization to the range [-1,1])

Step 2: Choice of initial set of features

Step 3: Feature selection - essential and non-essential features
*Criteria*: Use t-tests and scatter plots for each feature.

Step 4: For different combinations of features, train with adequate data set and hidden neurons.

Step 5: Choose the best possible set of features.
*Criterion*: Use the ability to reach the goal mean square error (MSE).

Step 6: Vary the number of hidden neurons as well as hidden layers for the chosen feature set.

Step 7: Choose the best networks for each group in terms of least number of neurons, ability to reach/go near the goal MSE.

Step 8: Final choice is based on test performance.
*Criterion*: generalization performance for test data

Step 9: Test performance for noisy data.

A three-layer neural network implementation in Matlab® environment is shown in Fig.3.13. In the present work, a three-layer network was attempted in a few cases (Appendix-E). However, for final implementations, only one hidden layer was needed, resulting in two-layer networks.

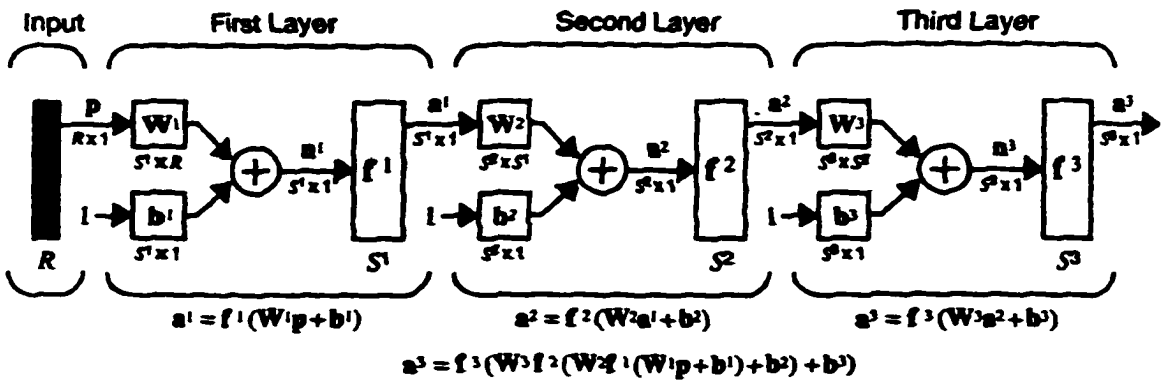



Fig. 3.13. NN structure implemented in Matlab® environment

A final remark on feed-forward neural network implementation is about the activation functions used. The present study used the *'tansig'* function for hidden neurons and *'purelin'* for output layers. The hidden layer with *'tansig'* activation function provided the required non-linearity to separate fuzzy feature space and 'purelin' functions in output layer just provide a simple linear transformation. Since weights and biases are the variables showing the training behavior, another set of variables are eliminated as biases are kept to zero.

### 3.8.4. Associative Network for Rhythm Classification

Detection of beat classes is followed by rhythm identification, which indicates the overall statement for the observed duration. For example, a rhythm statement for the following beats NOR-NOR-NOR-NOR-NOR-NOR should indicate normal sinus rhythm and a data segment with beats NOR-PVC-NOR-PVC-NOR-PVC should indicate bigeminy and so on. This task is essentially associating the information from beat classifier output to the rhythm class output. Such a task takes into account two

inputs: a code based on heart rate, and a set of codes corresponding to beats. The problem is similar to number identification in LCD displays.

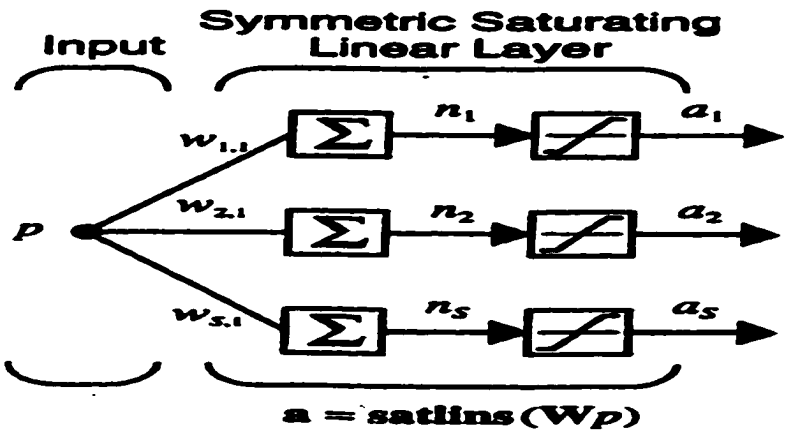



Fig. 3.14. Associative network for detecting the rhythm from beats

A typical rhythm associative network is shown in Fig. 3.14., and the associative network belongs to the class *Hebb's linear associator*. This problem is essentially a deterministic one with well-defined rhythm classes and a neural network just adds to the speed and fits with the overall design [Hagan *et al.*, 1995]. A sample-coding scheme implemented for the inputs to the associative networks is presented in Table. 3.3. The first four bits indicate the information regarding the heart rate; normal, bradycardia, and tachycardia. With thirteen beat classes detected in present scheme, a need arises for representing them with four bits. Hence, each beat class information requires four bits. The implementation is extendable, and for any new rhythm class, the new binary pattern has to be added in training stage. Binary codes applied in associative network implementation are specified below.

**Beat classes**: NOR-0000 RBBB-0001 LBBB-0010 VES-0011 PVC-0101
PACED-0110 APC-0111 ST/T-1000 JUN-1001 PREX-1010 AFIB-1011
AFL-1100 NOI-1101
**Heart rate classes**: NHR-0001 BC-0010 TC-0100

Table. 3.3. Sample input pattern for selected rhythm classes

| Original notation | Binary coding | Rhythm |
|---|---|---|
| NHR-NOR-NOR-NOR-NOR-NOR | 0001-0000-0000-0000-0000-0000 | NSR |
| NHR-NOR-PVC-NOR-PVC-NOR | 0001-0000-0101-0000-0101-0000 | VBIG |
| NHR-PVC-NOR-PVC-NOR-PVC | 0001-0101-0000-0101-0000-0101 | VBIG |
| NHR-PVC-PVC-PVC-NOR-PVC | 0001-0101-0000-0101-0000-0101 | Q* |
| TC-PVC-PVC-PVC-PVC-PVC | 0100-0101-0101-0101-0101-0101 | VTACH |

*-Unknown Combination-provide the input beat classification as output

As mentioned earlier, the associative memory learns to associate Q pairs of prototype input/output vectors. In other words, if the network receives an input $\mathbf{p}=\mathbf{p}_q$, then it should produce an output $\mathbf{a}=\mathbf{t}_q$, for q=1,2,....,Q. In addition, if the input is changed slightly (i.e. $\mathbf{p}=\mathbf{p}_q+\delta$), then the output should only be changed slightly (i.e. $\mathbf{a}=\mathbf{t}_q+\varepsilon$).

Associative networks learn by supervised Hebb's rule in the present work and the target output is known for each input vector. The resulting equation is

$$w_{ij}^{new} = w_{ij}^{old} + t_{iq} p_{jq} \tag{3.8}$$

where $t_{iq}$ is the ith element of the qth target vector $t_q$. Associative networks provide excellent approximations for nearest neighbor rules and tend to associate even an unknown input to one of the known patterns. In present task, there is no uncertainty assumed, and hence the task becomes considerably easier *i.e.,* a deterministic problem.

In implementation, an assumption is made about the availability of six beats. In cases of long data records or more number of beats, the first six beats were taken. On the other hand, in cases of bradycardia, a replication is done in order to get the six beat

class information. For example, in sinus bradycardia, NOR-NOR-NOR-NOR is taken as NOR-NOR-NOR-NOR-NOR-NOR and in ventricular bigeminy, NOR-PVC-NOR-PVC is taken as NOR-PVC-NOR-PVC-NOR-PVC, a replication for last two beats.

## 3.9. Evaluation Procedures

In the present work, testing and evaluation procedures are needed for each stage of the algorithm. Since the errors in initial pre-processing as well as feature extraction steps become cumulative in final classification, evaluation criteria are set very high. Evaluation procedures in the entire work can be divided into two stages; developmental and system.

Developmental evaluation can be viewed as comparing different methodologies and choosing the optimum one for a given task. System evaluation, on the other hand, evaluates the performance of the designed system and is helpful in fine-tuning the performance. Developmental evaluation, in the present context, deals with the following steps; pre-processing, fiducial point detection, and time and frequency domain feature estimation. System evaluation, on the other hand, deals with performance of the sub-systems after implementation, especially in the fiducial point detection system and the neural network-based pattern classification.

### 3.9.1. Developmental Evaluation

Developmental evaluation starts even from pre-processing steps. Sampling rate conversion is one clear example. In order to normalize the sampling frequency to a single frequency of Fs=250 Hz, the cubic interpolation re-sampling method was chosen based on the principle of least distortion. Nearly ten different filter and re-sampling

methods are analyzed. Results of the comparison are provided in chapter 4 [Srikanth *et al.*, 1998].

The major steps in fiducial point estimation are the detection of QRS complexes and subsequent R-points. Performance of QRS detection methodology in present work is very high (≈99%) and a detailed set of results for the entire data set is provided in next chapter. The performance of the ECG fiducial point estimation algorithms can only be compared with the performance of human experts. Fiducial point location is easier to evaluate, compared to the classification problem. Two cardiologists, as well as one graduate student verified the location and presence/absence of the fiducial points. The reference set is used for the error calculation for the fiducial point detections.

There are two stages in the fiducial point detection algorithm;

(i) Detection of waves alone (i.e.) P wave, QRS complex and T wave. A set of seven points are detected, namely P wave peak, Q start point, Q point, R point, S point, J point and T wave peak point.

(ii) Detection of starting and end-points for P and T waves to complete a set of nine points in any beat. Final set of fiducial points include P wave start, P wave end, Q start point, Q point, R point, S point, J point and T wave start and T wave end.

Evaluation procedures for fiducial point detection (FPD) algorithm make use of a set of 200 ECG recordings. Each ECG segment contains around 6-10 seconds of ECG information. The number of beats inside this interval varies. The algorithm is applied on all ECG data in the test set. Three factors are taken into account in deciding the adequacy of the algorithm. The factors include

(i) An ability to provide a result for the given input data (C1).

(ii) Accuracy in the location of the waves (C2).

(iii) Accuracy in the final location of the points by the algorithm (C3).

Criteria C1, C2, and C3 pose increasingly tougher problems for the algorithm. The algorithm is applied on the data set and the performance tested on all data. A table of results is presented after iteration on all the data segments and for visual recognition, plots are presented in the next chapter. Table. 3.4. presents a sample illustration. Such an approach provides a feedback on the efficacy of the modifications made and helps to avoid unneeded changes in thresholding.

Such a methodology resulted in seven revisions of the algorithm and the final form is found to provide more than 90% accuracy in exact identification of fiducial points (C3). The algorithm also satisfied the first criteria for all the data segments chosen (C1). The second criterion was satisfied for more than 98% of the waves (C2). Data segments showing huge errors are separated during the fiducial point detection stage and frequency domain techniques will be applied on such data. Such a strategy makes use of the errors in a positive way. Final numerical results are presented in chapter 4.

Table. 3.4. Evaluation of accuracy of estimated fiducial points - sample table

| Record | Attempts | Modifications made | Problem beats | Status |
|---|---|---|---|---|
| M1001 | 2 | J point elevated | 3,4 | Ok |
| M1003 | 3 | P and T wave overlap removed | 2,3,4,5 | Ok |
| M1059 | 3 | T wave boundaries | 1,2,3,4,5,6 | Ok |
| M1072 | 3 | One small P wave missed-detected | 3 | Ok |

3.9.2. System Evaluation

System evaluation in the current work specifically applies to the performance of the neural networks. Two different evaluation procedures are attempted. As suggested

by Heden, 1996, in his dissertation, around one quarter of the original data is available for testing. The initial testing is however with respect to the training data and how well the network adjusts for the set.

Testing is also continued further with a newer set of data, not exactly similar to the training data. A few cases of simulated beats provide an ideal set of test data. Depending on the performance, some of the training steps might be implemented again. In the present case, random noise is added to a collection of ECG signals and fed as input to the system for evaluation. Gaussian random noise is generated using Matlab®-simulink and the amplitudes are normalized with respect to the amplitude of the ECG signal. Hence, three sets of evaluation results are available for the selected network, for different sets of inputs.

(i) Training data,

(ii) Test data, and

(iii) Noisy data.

Feature extraction provides a very good buffer, with respect to the disturbance and noise in the input. It eliminates worry about these aspects and non-stationarity, and provides the normalized values of the standard features as input to the network. Performance with respect to noisy data is evaluated at the level of fiducial point detection and feature extraction and also at final beat classification task. As the performance of the overall system mainly depends on the accuracy of the features, the presence of outliers in feature space warrants rejection in the present work.

## 3.10. Comments on the Design and Methodology

Design of a computerized ECG interpretation module derives ideas from several areas and the block diagram in Fig.3.1. illustrates the point. Providing proper interfaces between the blocks and optimizing the performance of each block are the two major goals in the current design. Errors in each block tend to be cumulative and care should be exercised in controlling the error to a minimum. Chapter 4 provides results in implementation of individual blocks as well as the final neural network based classification scheme.

# CHAPTER 4

# RESULTS

## 4.1. Introduction

A single-lead electrocardiogram classification scheme using artificial neural networks (ANN) is essentially a multi-component system, involving several sub-systems. The bottom-up approach to design involves perfecting the performances at sub-system level with the objective of improving the overall system performance. Hence, the need arises for evaluating each individual component in the system. This chapter presents performance results at the sub-system level and the overall system level of implementations. Results are analyzed for individual systems like pre-processing, fiducial point identification, feature extraction, and the final pattern classification using ANNs.

## 4.2. Results in Pre-processing

Pre-processing in the present work consists of following tasks: manual annotations for individual beats, band-pass filtering of the signal, and amplitude, and sampling frequency normalizations. Classification of beats is already available in MIT-BIH data set. A team of three cardiologists and two graduate students assisted in the final classification of clinical data. Each of the cardiologists had more than ten years of clinical experience and graduate students were chosen based on their previous clinical

experience. The results and the optimal implementations are explained in following sub-sections.

4.2.1. Low-Pass Filters

Low-pass filters are designed with several alternative structures and designing digital filters is simple due to the Matlab® toolbox. The final low-pass filter has a cut-off of $f_c$=100 Hz and is applied to both the MIT-BIH data and clinical data after de-trending. The filter is an IIR filter of Chebyshev-Type 1 of sixth order. This frequency range is able to avoid baseline drift, while maintaining the frequency content of major components in the signal. The final frequency ranges for MIT-BIH data (0.1-100) Hz and clinical data (0.04-100) Hz both satisfy the prescribed norms. [Moody, 1992; Suzuki and Ono, 1992; MacFarlane and Lawrie, 1974]. Base-line wander remained in the signal and is included along with noise as a class. Figures 4.1-4.3 show the effect of a higher cut-off for high-pass section at $f_c$=0.5 Hz.

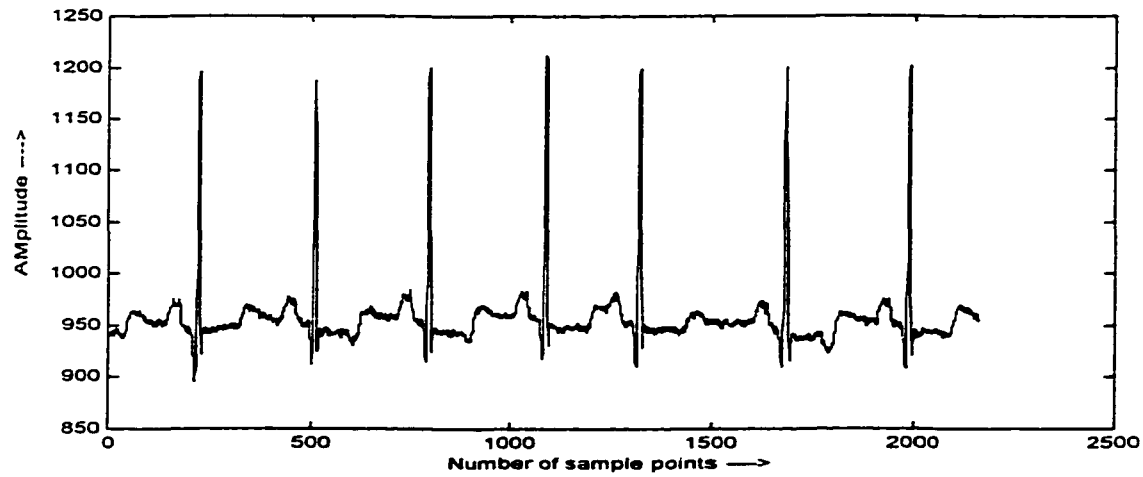



Fig. 4.1. Original Signal (MIT-BIH) (0.1-100) Hz - analog band-pass

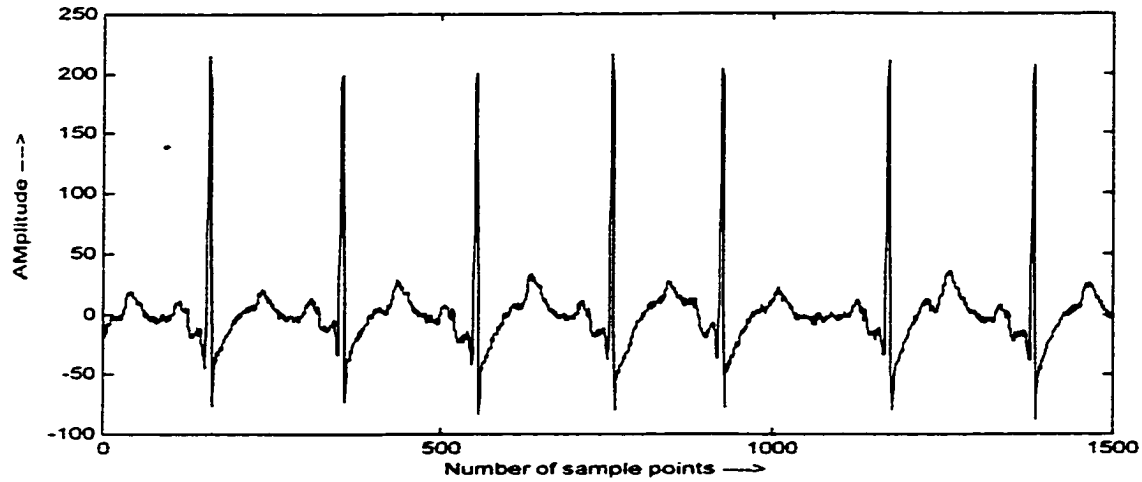



Fig. 4.2. Output for Bandpass Filter (0.5-100) Hz

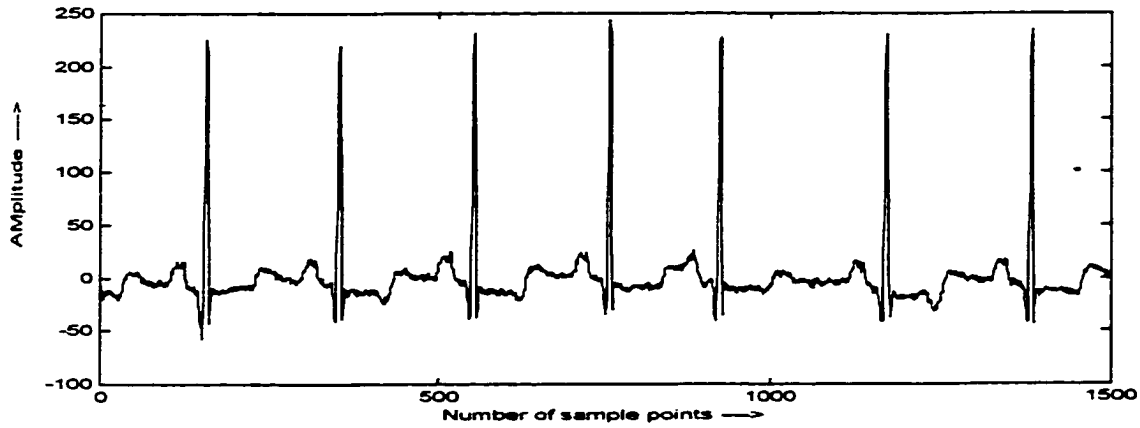



Fig. 4.3. Output of lowpass filter-Fc=100 Hz - signal used in present thesis

4.2.2. Resampling Techniques

Developing new applications in neural networks related to ECG pattern recognition problems requires large, well-tested databases. In the present work, data are extracted from the MIT-BIH database, and from clinical subjects. Some ECG data segments, used in development only, are also generated by simulation. Sampling frequency for data from different sources vary between 200 to 500 Hz. Normalization of sampling frequency to a single value is vital in developing and testing new signal processing algorithms. All signals are converted to a single sampling frequency of 500

Hz in order to preserve maximum information. A comparison of performances of different algorithms for the conversion of sampling rates for ECG signals resulted in the evaluation of eight different methods.

The methods can be divided into four major classes, namely those following (i) polynomial interpolation, (ii) computation in digital domain, (iii) frequency domain techniques, and (iv) digital re-sampling filters. Linear, cubic, and spline interpolations belong to polynomial interpolation class. Backward step and nearest neighbor interpolations are the computations in digital domain. In frequency domain techniques, interpolation is performed in the frequency domain, and the result is transformed back into the time domain. Digital filters include both infinite impulse response (IIR) and finite impulse response (FIR) structures. Performance evaluation involved the estimation of time and frequency domain features for the signal. Estimation of features before and after re-sampling revealed the ability of different methods to preserve the original waveform. Frequency domain features seem to be more sensitive compared to time domain features, in highlighting the minor changes taking place after re-sampling. ECG beats before and after re-sampling are shown in Fig. 4.4.(a)-(c).

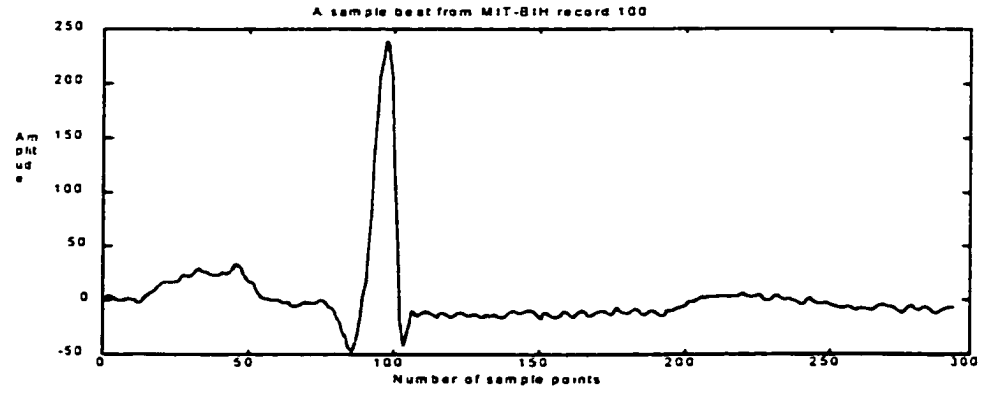



Fig. 4.4.(a). A sample beat from MIT-BIH data base record 100 -- Fs=360 Hz

Fig. 4.4. Effect of re-sampling on ECG waveforms

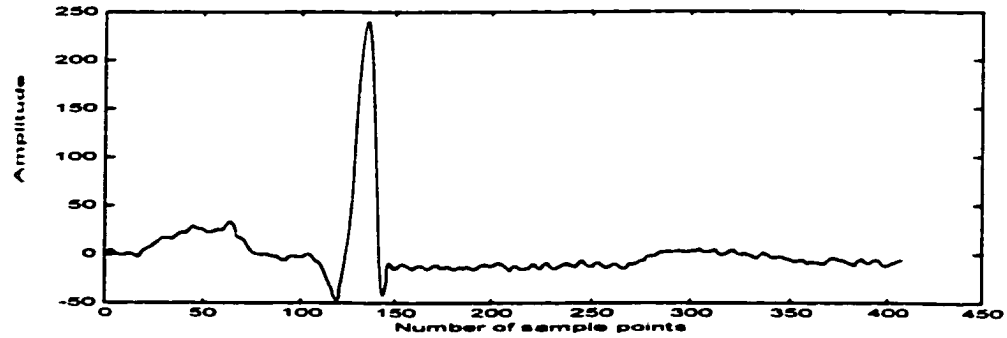



Fig. 4.4.(b). Cubic interpolation of above beat to Fs=500 Hz

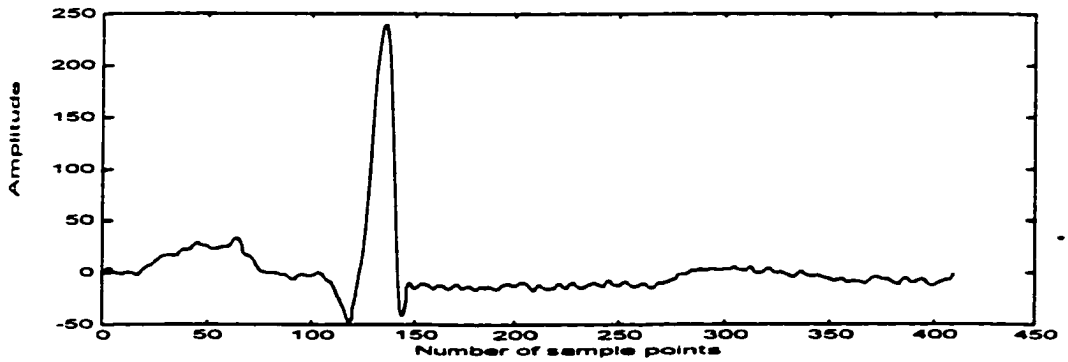



Fig. 4.4.(c). FIR-Kaiser window polyphase resampling filter for Fs=500 Hz
Fig. 4.4. Effect of re-sampling on ECG waveforms (Cont.)

Table. 4.1. shows the performance of the eight techniques on the beats from record 100 in MIT-BIH database [Srikanth *et al.*, 1998]. Original signal sampling frequency is 360 Hz and the new sampling frequency is 500 Hz. Based on numerical values of the features as well as visual criteria, polynomial interpolation methods seem to perform better in terms of simplicity in implementation and the criterion of least distortion. Numerical features in frequency domain provided information on the quality of the re-sampling and distortions introduced in the process. Visual criteria included the observation of effect on end points. Hence, cubic interpolation is chosen as the method of choice in present work.

Table.4.1. Feature values for a segment of six seconds duration from MITDB 100 record

| Features / Methods | Abnorm Points | PSD ratio1 (0.5-6) Hz | PSD ratio2 (6-12) Hz | PSD ratio3 (12-18)Hz | PSD ratio4 (18-30)Hz | IF (Hz) | MPF (Hz) | Dbcutoff (Hz) |
|---|---|---|---|---|---|---|---|---|
| Original | 0 | 0.191 | 0.257 | 0.248 | 0.170 | 14.57 | 6.25 | 57.250 |
| Linear | 19 | 0.194 | 0.260 | 0.248 | 0.169 | 14.37 | 6.25 | 57.250 |
| Nearest | 8 | 0.191 | 0.258 | 0.247 | 0.169 | 14.91 | 6.25 | 61.000 |
| Cubic | 20 | 0.191 | 0.258 | 0.248 | 0.170 | 14.52 | 6.25 | 57.250 |
| Spline | 19 | 0.191 | 0.257 | 0.248 | 0.170 | 14.54 | 6.25 | 57.250 |
| Freq. Domain | 19 | 0.191 | 0.257 | 0.248 | 0.170 | 14.56 | 6.25 | 60.500 |
| Backward step | 8 | 0.192 | 0.257 | 0.248 | 0.168 | 14.89 | 6.25 | 61.750 |
| IIR filter (Analog) | 70 | 0.109 | 0.182 | 0.210 | 0.098 | 21.98 | 11.0 | 110.25 |
| FIR (Kaiser) | 22 | 0.184 | 0.254 | 0.243 | 0.167 | 16.52 | 6.25 | 66.750 |

## 4.3. Results of QRS Detection

Implementation of QRS detection algorithm in present work is explained in chapter 3. The implementation results in a very high accuracy in QRS complex detection and R point estimation. Table.4.2 presents the results from QRS detection, tested with the overall set of 3734 beats selected from both MIT-BIH database and clinical data. A small number of simulated beats are also tested.

Table.4.2. Statistics on QRS detection

| | |
|---|---|
| Total number of beats | 3734 |
| Proper R point detection | 3711 |
| Noise reduction after double smoothing for detected beats (T14)* | 1.02-1.20 |
| Noise reduction after double smoothing for discarded beats (T14)* | 1.16-1.32 |
| Number of beats classified as noisy beats | 242 |
| Noise reduction after double smoothing for these noisy beats (T14)* | 1.16-1.34 |

*Explained in chapter 3

## 4.4 Results of Overall Fiducial Point Estimations

Performance of overall fiducial point estimation methods is always dependent on QRS detection efficiency. Figures 4.5.(a)-(h), illustrate a variety of problems associated with both QRS detection as well as overall fiducial point identification. The exact quantity providing the numerical accuracy of fiducial point detection algorithms is the number of discarded beats, due to abnormal value in features (Table3.1.). Total number of discarded beats constitutes just 1.75% of the total of 3734 beats, providing an accuracy of about 98.25%. An accuracy of 98.25% is quite excellent, considering the choice of beat classes include atrial flutter, atrial fibrillation, ventricular beats and extremely noisy beats. In the present work, the accuracy of detected fiducial points is decided by an ability to estimate feature values for further classification. This criterion is not as strict as the exact location criteria suggested by others [Laguna *et al.*, 1994; Greenhut *et al.*, 1989], but an attempt is made to incorporate the expert feedback on exact locations as explained in chapter 3.

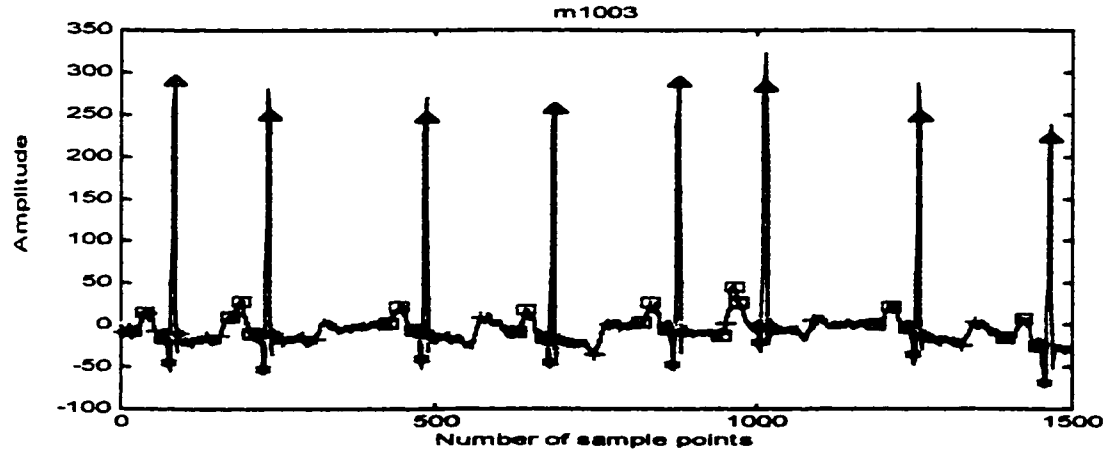



Fig. 4.5.(a). Errors in T wave identification for atrial premature beats
Fig. 4.5. ECG Segments with detected fiducial points marked

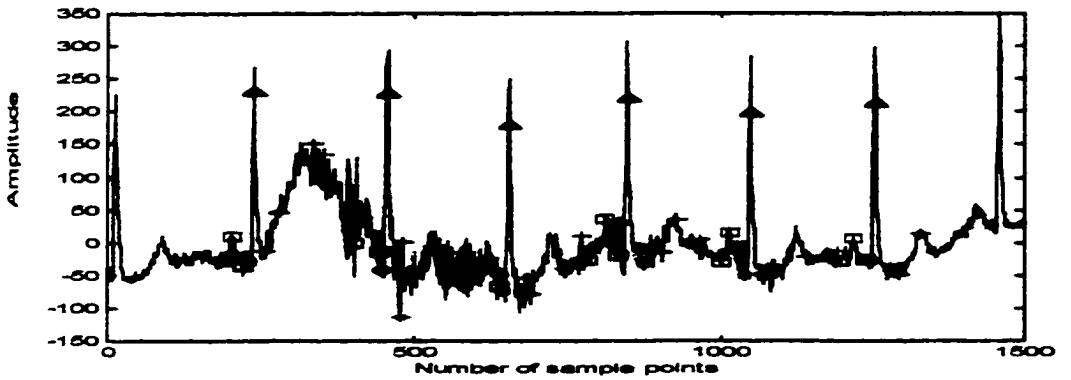



Fig. 4.5.(b). Fiducial point extraction in the case of noisy segment

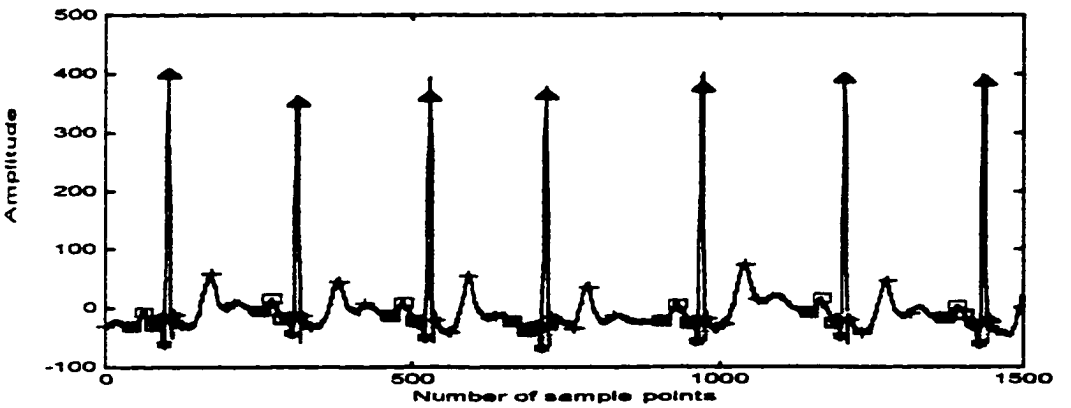



Fig. 4.5.(c). Fiducial point detection in a simple data segment, with no complexities

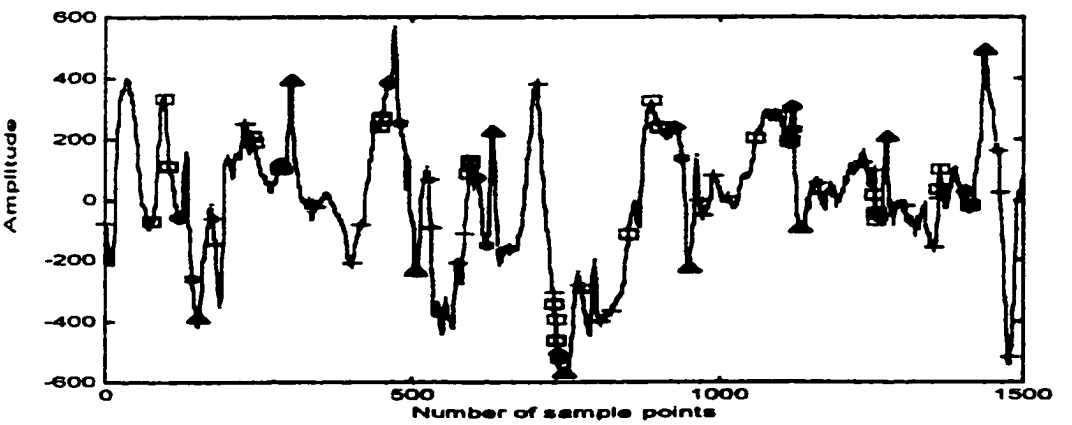



Fig. 4.5.(d). A case of signal corrupted by large noise -- one of the discarded segments
Fig. 4.5. ECG Segments with detected fiducial points marked

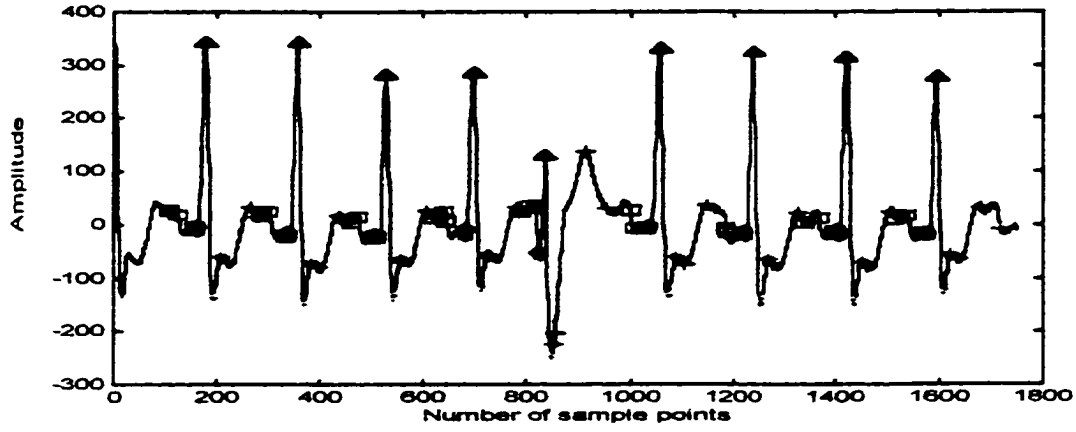



Fig. 4.5.(e). Fiducial point detection in PVC beat in the middle of LBBB beats

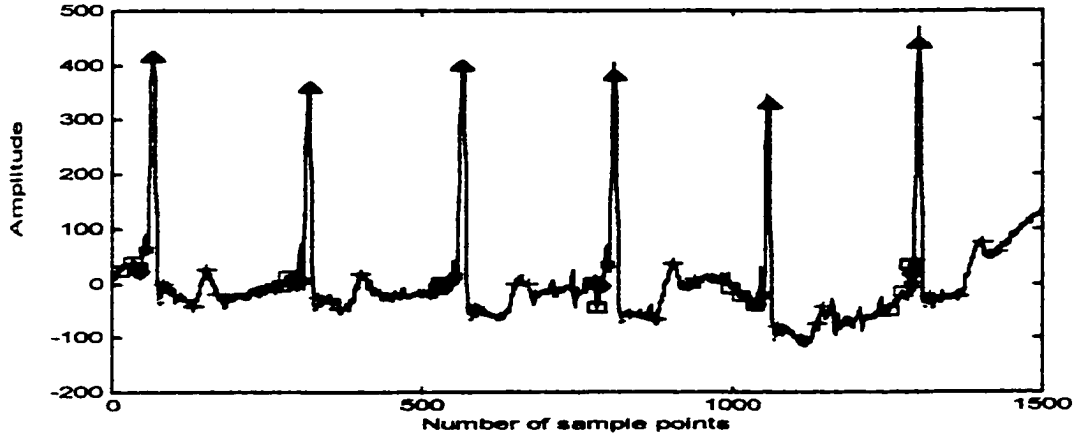



Fig. 4.5.(f). Fiducial point detection in case of APC and JUN beats

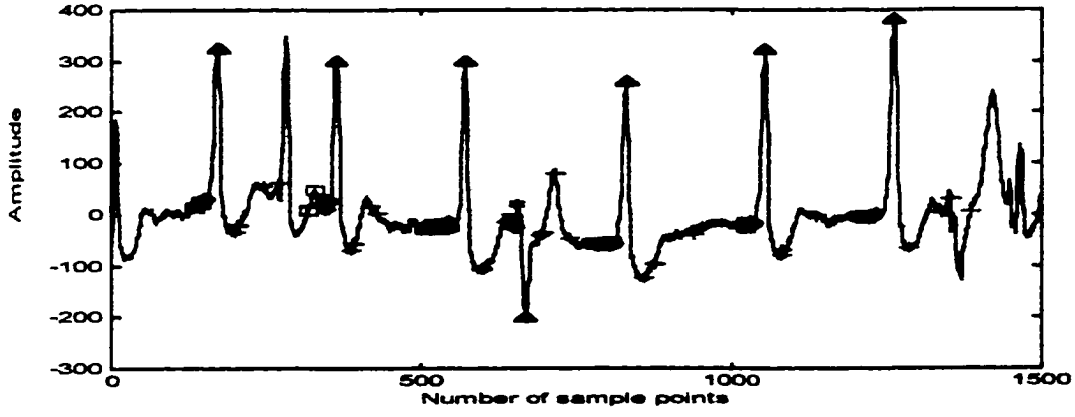



Fig. 4.5.(g). Fiducial point detection in AFIB beats -- approximate $P_1$ location is enough

Fig. 4.5. ECG Segments with detected fiducial points marked

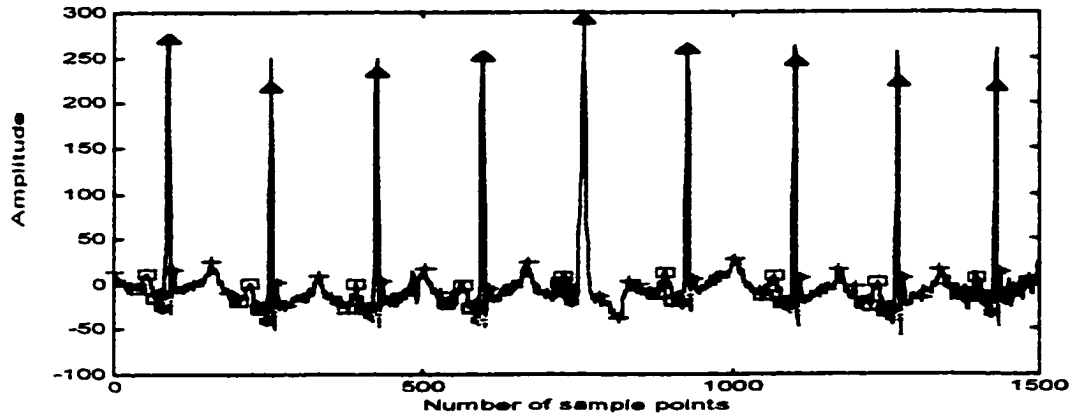



Fig. 4.5. (h). Fiducial point identification in a noisy segment

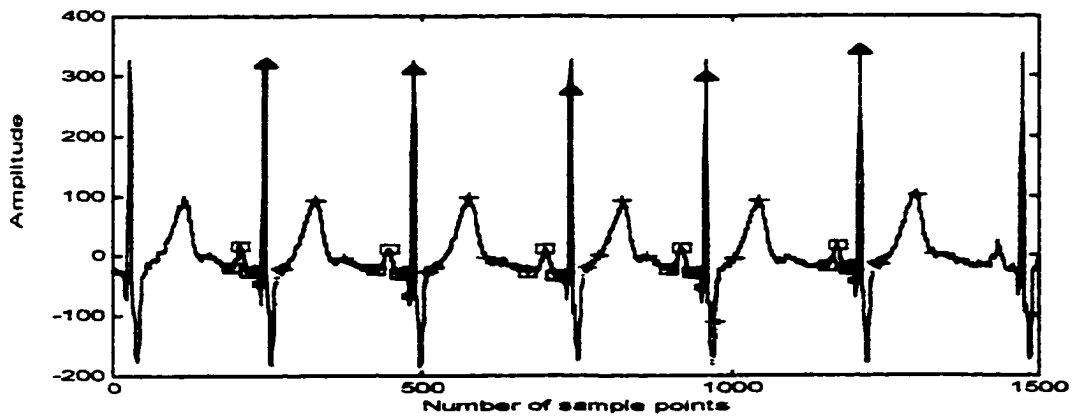



Fig. 4.5. (i). Fiducial point detection in a simple right bundle branch block segment

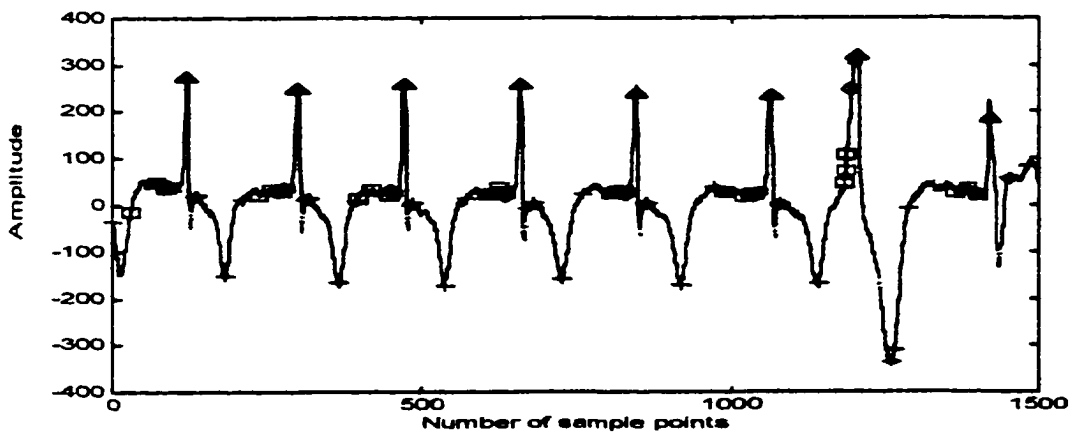



Fig. 4.5.(j). Fiducial point detection in atrial fibrillation with inverted T waves
Fig. 4.5. ECG Segments with detected fiducial points marked (Cont.)

Fiducial point detection algorithms are evaluated for the entire data segments as well as individual beats. Only three data segments out of initial 498 segments were discarded due to entirely bad detection, and a sample is shown in Fig. 4.5.(d). Errors in other segments vary from no errors to errors in multiple beats. In cases of atrial fibrillation, atrial flutter and ventricular beat, approximate detection of start and end points of a beat are enough to estimate features in the frequency domain. Interestingly, these algorithms make use of syntactic, hard-threshold logic in detecting peaks and boundaries and show glaring deficiencies in location of fiducial points in some cases, as shown in Figures 4.5. (a),(f) and (j). However, the location of fiducial points is more of a linearly separable problem and hence the accuracy achieved is quite high, eg: separation of beat boundaries.

## 4.5. Results of Selection of Features

In this section, the results of time and frequency domain features and the range of their variations are presented. The values of means, maximum, minimum and standard deviations of individual features are presented in Table.4.3. Definitions of the features in both time and frequency domains are presented in Chapter 3. Selection of features in a particular group is based on classification information provided by the feature set. Beats with physically impossible feature values are discarded. For example, negative interval value calculations, indicate the errors in fiducial point estimation and such beats are discarded. Estimated error rate of 1.75% of the beats in whole data set is very good, especially since difficult classes like APC, AFIB, AFL and NOI constitutes more than 25% of the set. Further, the inputs to the neural networks are

screened for outliers and feature values deviating beyond six times the standard deviation are eliminated from training and testing sets.

Table. 4.3. Range of values of features in time and frequency domains

| Feature (unit) | Mean | Maximum | Minimum | Standard deviation |
|---|---|---|---|---|
| T1 (sec) | 0.7242 | 1.5960 | 0.2720 | 0.1927 |
| T2 (scaled amplitude) | -4.8948 | 289.9429 | -193.4404 | 30.7066 |
| T3 (categorical) | 0.3803 | 2 | 0 | 0.6083 |
| T4 (ratio) | -1.2883 | 2684.2 | -2524.7 | 99.88 |
| T5 (sec) | 0.0419 | 0.3280 | 0.0040 | 0.0502 |
| T6 (scaled amp /sec) | 0.2493 | 27.005 | -24.639 | 3.1153 |
| T7 (sec) | 0.1359 | 0.4200 | 0.0080 | 0.0634 |
| T8 (categorical) | 0.3884 | 2 | 0 | 0.7290 |
| T9 (sec) | 0.1343 | 0.9880 | 0.0080 | 0.1479 |
| T10 (sec) | 0.1088 | 0.4120 | 0.0080 | 0.0614 |
| T11 (ratio) | -0.9495 | 147.6 | -31.2408 | 2.6600 |
| T12 (ratio) | -0.9072 | 8566.6 | -11051.0 | 344.1334 |
| T13 (sec) | 0.4540 | 0.980 | 0.1640 | 0.0919 |
| T14 (ratio) | 1.1362 | 4.400 | 0.4480 | 0.3238 |
| T15 (count) | 43.9193 | 198.0 | 1 | 21.6981 |
| T16 (sec) | 0.1425 | 3.8240 | 0.040 | 0.1939 |
| F1 (Hz) | 7.6565 | 28.5214 | 0.1424 | 3.3440 |
| F2 (ratio) | 0.4719 | 0.9147 | 0.0415 | 0.1789 |
| F3 (ratio) | 0.2408 | 0.5416 | 0.0039 | 0.0974 |
| F4 (ratio) | 0.1226 | 0.4002 | 0.0006 | 0.0735 |
| F5 (Hz) | 8.1751 | 19.8140 | 1.0078 | 2.7633 |
| F6 (Hz) | 4.2801 | 7.0995 | 0.3342 | 1.2344 |
| F7 (Hz) | 13.0860 | 17.0260 | 10.5329 | 1.2606 |
| F8 (Hz) | 23.2593 | 26.0140 | 20.8863 | 0.7118 |
| F9 (ratio) | 0.4293 | 0.8995 | 0.0593 | 0.1666 |
| F10 (ratio) | 0.2979 | 0.7393 | 0.0111 | 0.1518 |
| F11 (ratio) | 0.1283 | 0.4313 | 0.0062 | 0.0840 |

4.5.1. Results of Frequency Domain Feature Estimations

Chapter 3 presented six different spectral estimations attempted in a single beat analysis. The need arises for eliminating estimations that provide the least information to reduce the computational burden. A small data set of 350 beats was used to classify ten different classes using a rule-based classifier (VP-Expert®) shell. The sample beats and their classes are clearly identified and the sensitivity of the estimations are calculated based on visual criteria and scatter plots for feature space. Distribution of the feature values for different classes provide an indication of the difficulty in separating them with the least error. The final decision is also based on accuracy needed in location of fiducial points for a given class, as shown in Table.4.4. Based on these results, the choices are restricted to the first two estimations, namely *est1* and *est2*, *i.e.*, spectrum of the original beats and the spectrum of the beats after subtraction of a simulated normal beat. These two estimations together provide a set of features sensitive to a wide array of classes, including noise and atrial flutter and atrial fibrillation.

A final set of eleven features in frequency domain is calculated from the above two estimations. Figure 4.6.(b) illustrates all six estimations for the first beat in ECG segment shown in Fig. 4.6.(a). Figure 4.7.(a) shows an ECG segment with inverted T wave morphology in all beats and corresponding spectral estimation after the subtraction of a simulated normal beat (*est2*) is shown in Fig. 4.4.(b). Figures 4.5 and 4.6 illustrate the clear distinction shown by *est2* for PVC beats compared to other classes. All figures have number of beats in X-axis, except the first one where the

spectrum is drawn for different estimations for a single beat. The Y-axis represents frequency domain axis and Z-axis represents power spectral density.

Table. 4.4. Sensitivity of spectral estimations

| Estimation | Sensitive Classes | Needed accuracy in fiducial point location -- Categorical |
|---|---|---|
| Est1* | NOR, PVC, PACED, NOI, AFIB, AFL | Normal accuracy enough |
| Est2* | NOR, RBBB, LBBB, AFL, PVC, PACED | Normal accuracy enough |
| Est3* | APC, PVC | Very high accuracy needed |
| Est4* | RBBB, LBBB, AFL, AFIB | Normal accuracy enough |
| Est5* | ST/T, PVC | Very high accuracy needed |
| Est6* | NOI, PACED, PVC | Very high accuracy needed |

**Est1* -- Spectrum of the original beat
**Est2* -- Spectrum of (original-simulated normal) beat
**Est3* -- Spectrum of (original beat-P wave)
**Est4* -- Spectrum of (original beat-QRS complex)
**Est5* -- Spectrum of (original beat-T wave)
**Est5* -- Spectrum of (original beat-(P+QRS+T) waves)

## 4.6. Results of Final Beat and Rhythm Classification

Feed-forward artificial neural networks (ANN) require an extensive and well-classified ECG database for both training and testing. The number of beats selected and other statistics regarding the database are presented in chapter 3. Only a few beats were discarded, due to physically impossible values or when they lie outside the normal range of variations. Such cases represent the complexity of the real, clinical data and in some cases, the failure rate of fiducial point detection algorithms. Initial neural network implementations provided clues to the required training data set for adequate generalization. Appendix-E provides a detailed account of the total number of data used in attempted training, features selected for training, and structural details of the

networks. Fig.4.10 presents the structure of neural network implementation. The current section is continued in page 118.

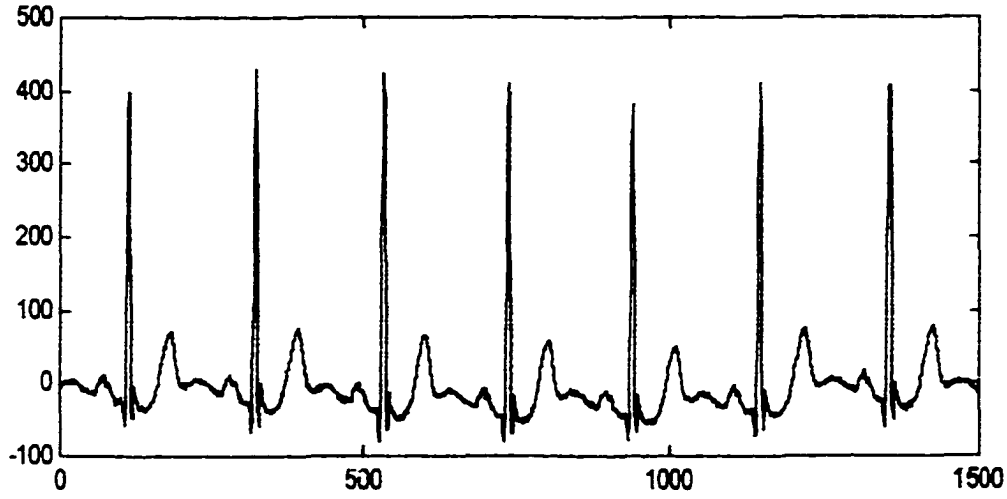



Fig.4.6.(a). An ECG segment with normal sinus rhythm beats

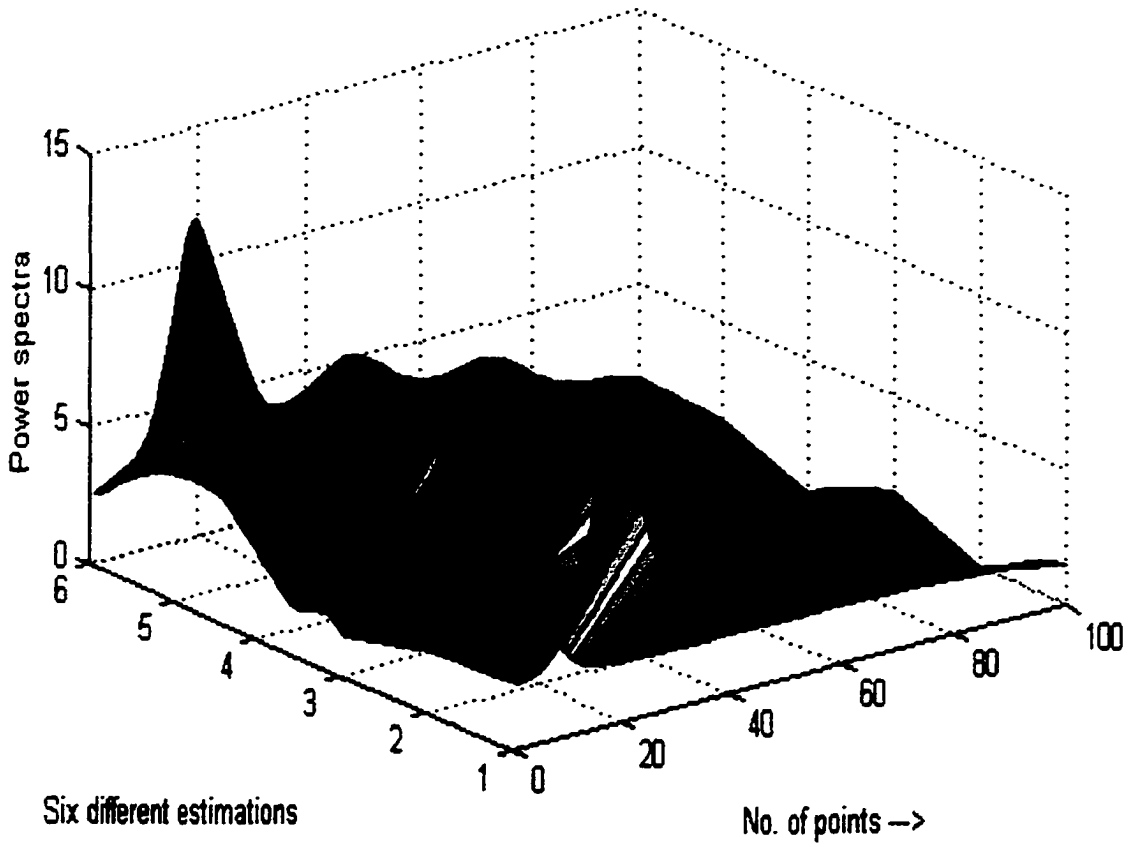



Fig.4.6.(b). All six estimations shown for the first beat in the above figure

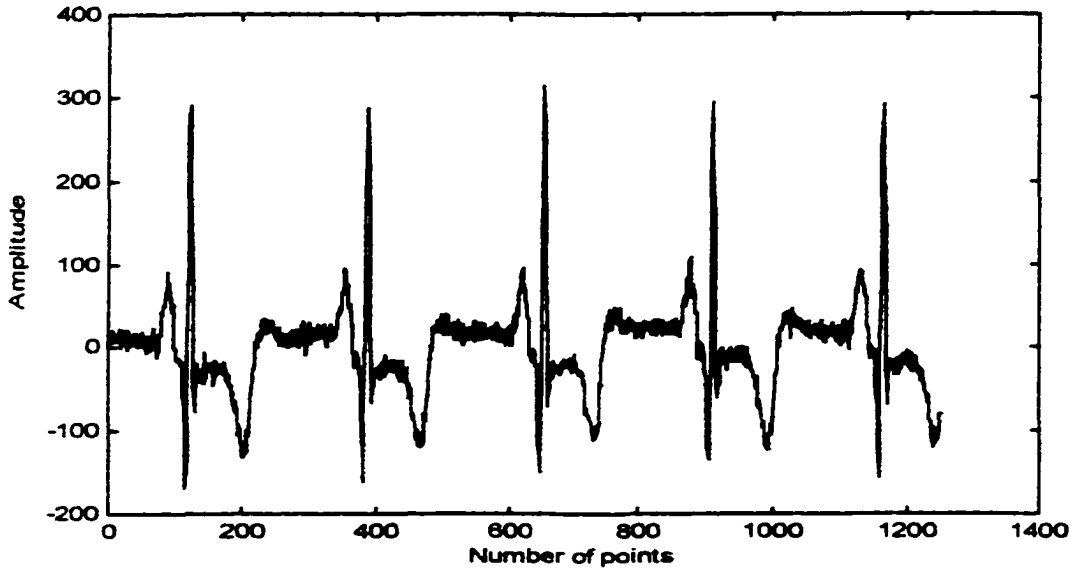



Fig.4.7.(a). A sample ECG segment with inverted T wave morphology

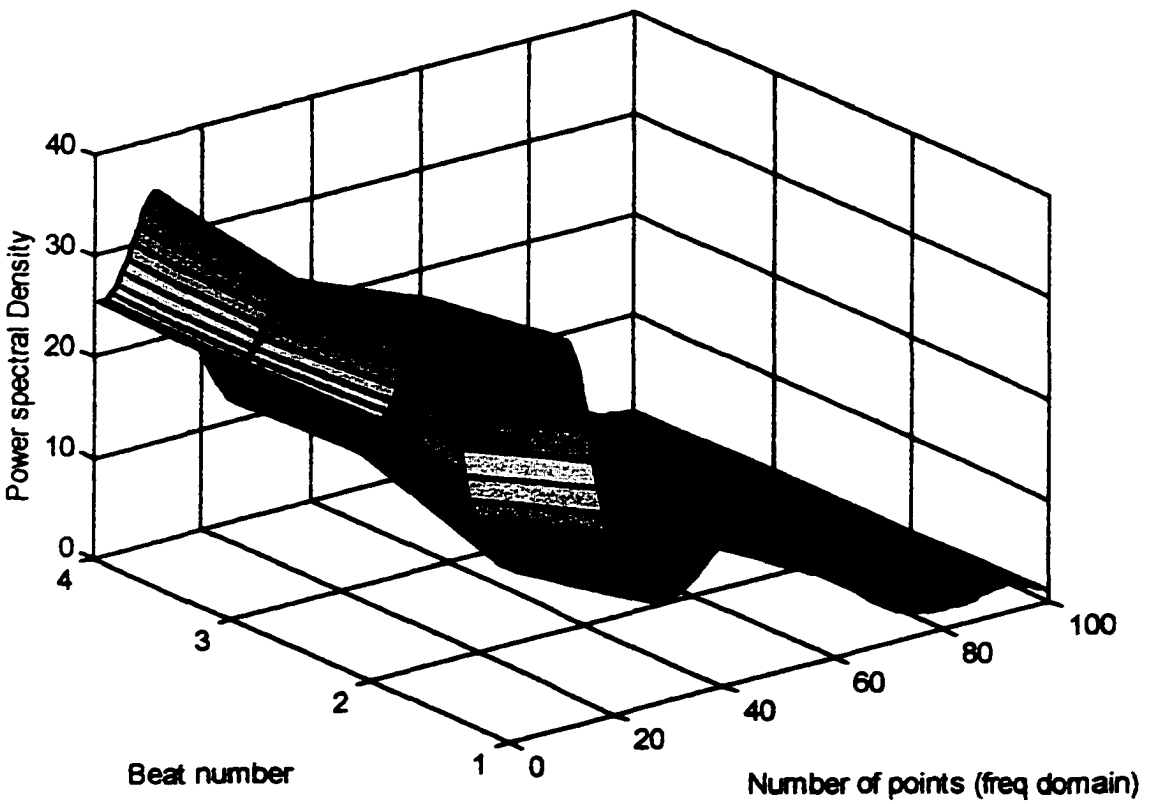



Fig. 4.7.(b). Spectrum of *disturb(t)=beat(t)-nsr(t)* for all beats in above ECG segment

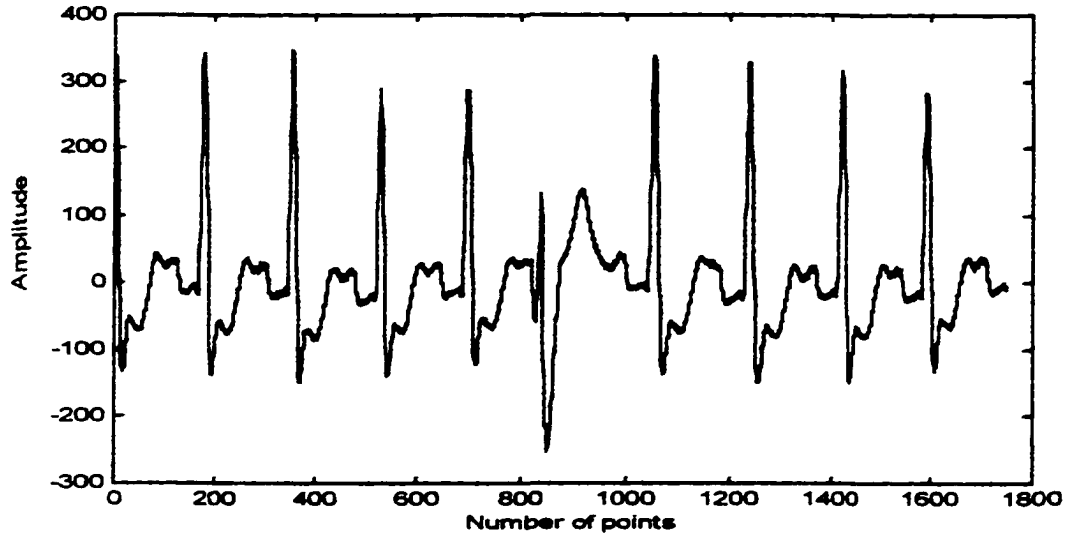



Fig.4.8.(a). ECG data segment with combination of LBBB and PVC beats

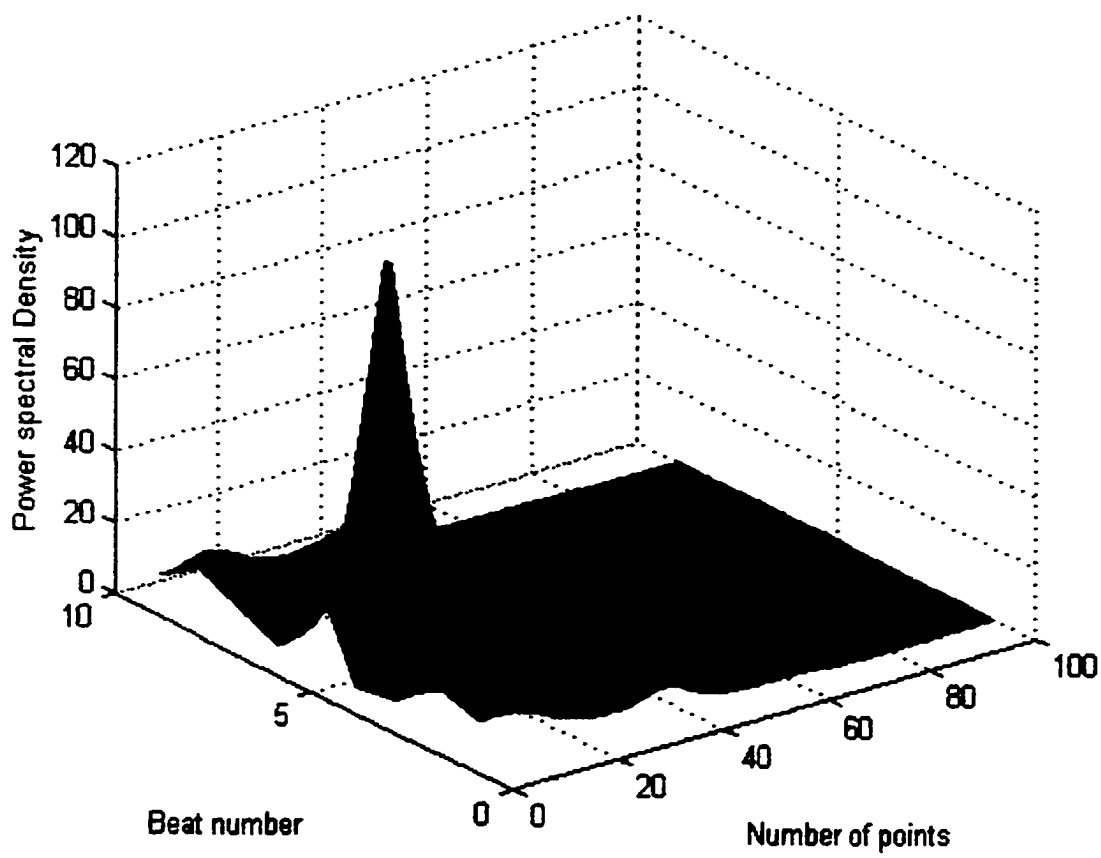



Fig. 4.8.(b). Spectrum of *disturb(t)=beat(t)-nsr(t)* for all beats in above ECG segment

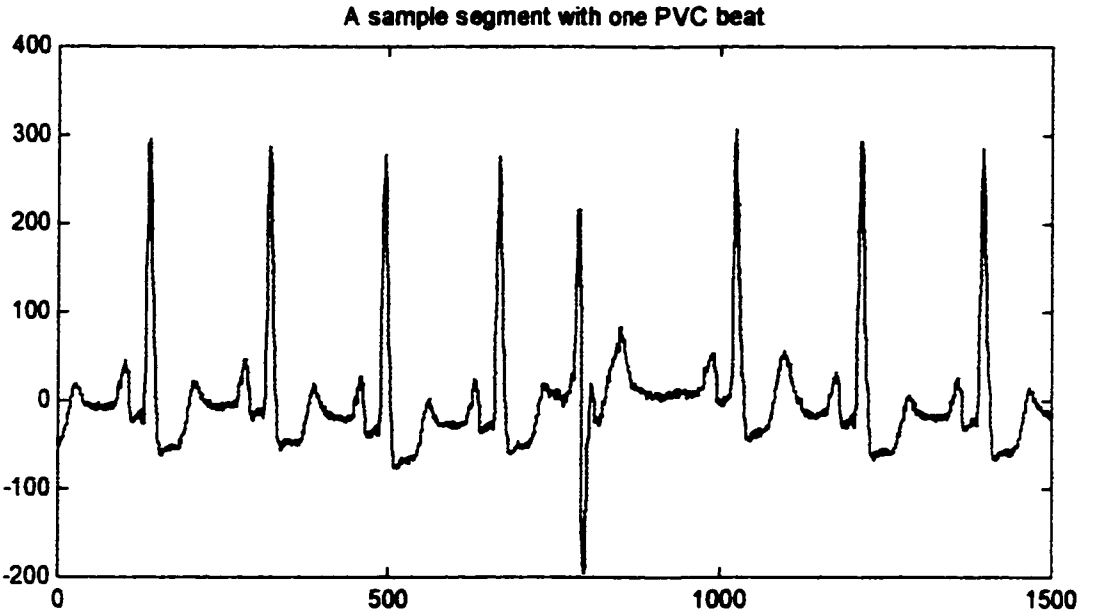



Fig.4.9.(a).Sample ECG segment with one PVC beat

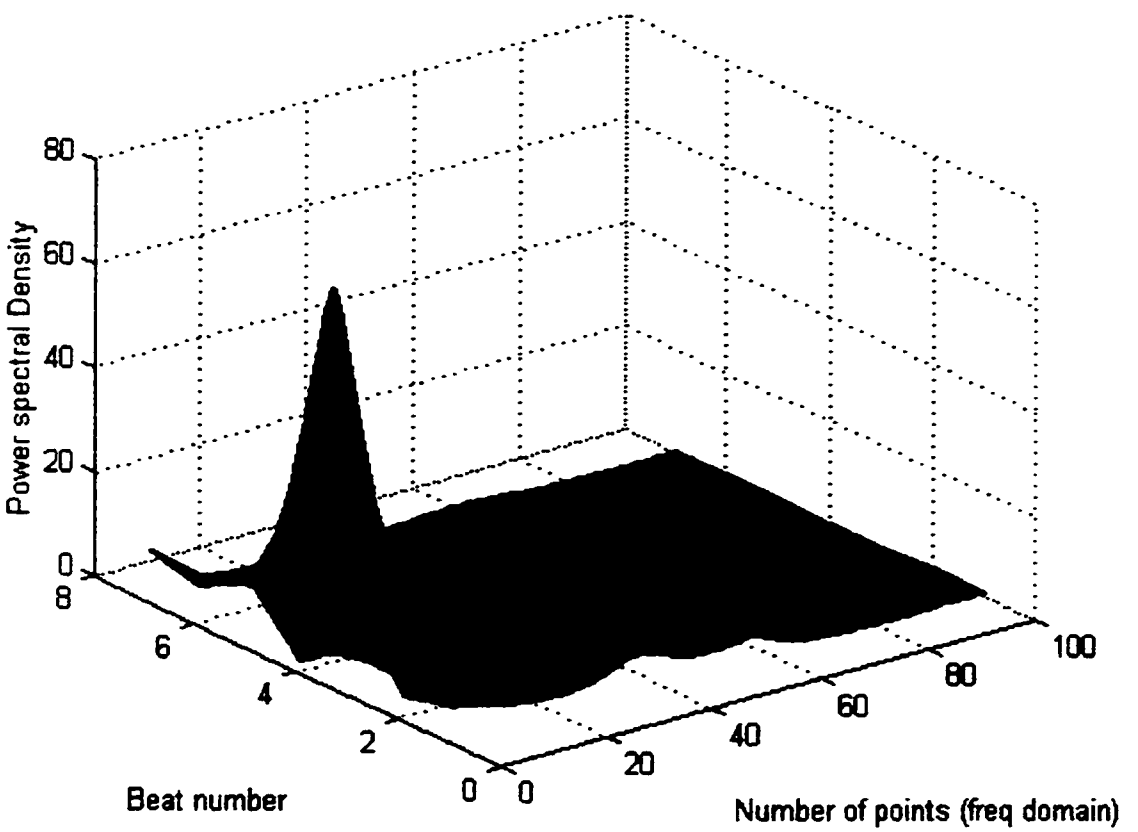
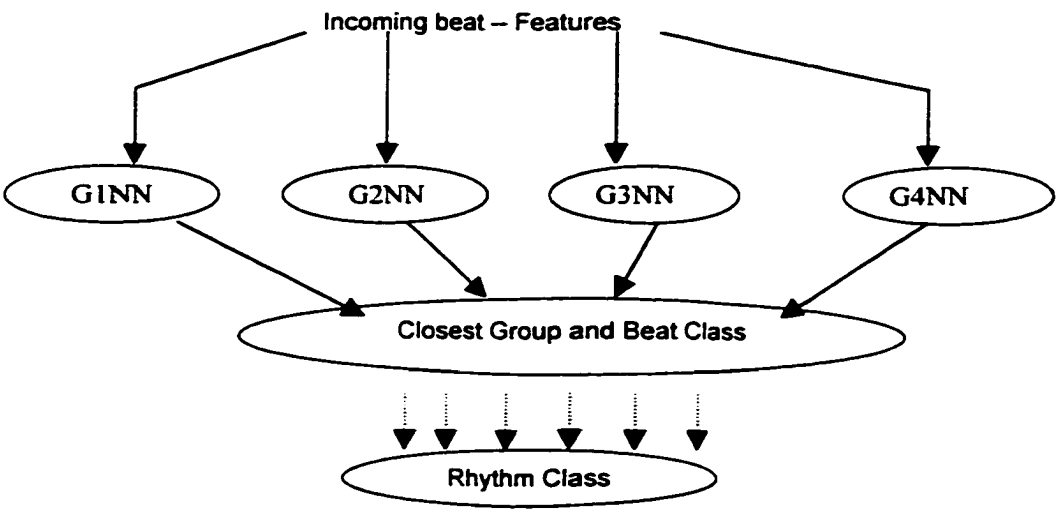



Fig. 4.9.(b). Spectrum of *disturb(t)=beat(t)-nsr(t)* for all beats in above ECG segment




*G1NN, G2NN, G3NN, and G4NN -- Feed forward neural networks to classify the member classes in each group*

Fig. 4.10 Overall neural network implementation of the classification

Implementation of ECG beat and rhythm classification with neural networks in present work involves analyzing four steps: choice of an optimal network, beat classification results, testing with noisy data and rhythm classification results.

4.6.1. Choice of Optimal Network

Choice is based on the performance of the networks in training and testing. Criteria include both training criteria and testing criteria. Training criteria include

(i) An ability to reach the minimum value of mean square error (MSE)

(ii) Small number of hidden neurons and

(iii) Small number of features.

Testing criteria include

(i) An ability to generalize for test data and

(ii) An ability to generalize for noisy features.

Training and testing data sets are chosen adequately large, so that each epoch contains a minimum of 5-10 times the number of connections between the input and hidden neurons. Appendix-E provides the entire set of neural networks tested with data sizes for training. Details included in the Tables in Appendix-E are the structure of the particular neural network, number of epochs in training, training goal in terms of mean square error, and the mean square error achieved after training. Neural networks varying in the number of hidden neurons, input features and structure are all tested for their ability to reach minimum mean square error goal. Stability in mean square error values are also noted. Feature sets are based on combination of the available features with some essential features for each group. Networks satisfying the minimal standards in training are chosen for testing. Some of the selected networks are found to lack generalization capability. The reason for such cases is that the training is based on a very small data set and hence, the network has to be re-trained with a much larger training set. Details of the size of the smaller data set are provided below the Table. Individual feature sets for each neural network are not shown and are provided only for the final implementations. Design became more systematic, as the problem moved from the first group to the later groups. The next section presents the structure, training and testing performances of the selected networks. Two networks are selected for groups one, three and four and four networks are selected for the second group.

4.6.2. Results in Beat Classification

Table.4.5 presents the performance of chosen networks for each group and their performances in testing. Problems encountered in this step included

(i) Inadequate samples for few classes, dominant major class in each group

(ii) Training did not involve a thoroughly mixed data in each epoch

Performance is defined as percentage of the training and testing cases properly classified. Table.4.6 presents the results after overcoming these two problems. A few minority beat classes were added in training and a thorough randomization is performed before each epoch. The total number of training and testing data are kept same for comparison. Beat classes with inadequate number of samples, *i.e.*, ventricular escape beats (VES), Wolf-Parkinson-White (WPW) pre-excitation beats, Nodal beats (JUN) and atrial flutter (AFL) beats, are supplemented with 40, 20, 30 and 20 more beats respectively. These additional beats were extracted from MIT-BIH data records.

Table.4.5. Results for feed-forward neural networks in each group with inadequate minority sampling

| Group | Network | Structure | Training data | Performance | Test data | Performance |
|---|---|---|---|---|---|---|
| Group 1 | Nnet11 | [7,18,3] | 701 | 88.45% | 707 | 74.68% |
| | Nnet12 | [7,20,3] | 701 | 88.45% | 707 | 71.71% |
| Group2 | Nnet21 | [7,4,4] | 560 | 87.50% | 187 | 86.10% |
| | Nnet22 | [7,5,4] | 560 | 88.57% | 187 | 89.30% |
| | Nnet23 | [7,6,4] | 560 | 89.64% | 187 | 87.70% |
| | Nnet24 | [7,7,4] | 560 | 91.25% | 187 | 84.49% |
| Group3 | Nnet31 | [7,8,3] | 509 | 90.96% | 168 | 85.12% |
| | Nnet32 | [7,9,3] | 509 | 91.55% | 168 | 82.74% |
| Group4 | Nnet41 | [8,15,3] | 531 | 84.18% | 173 | 64.23% |
| | Nnet42 | [8,17,3] | 531 | 87.01% | 173 | 69.74% |

Table.4.6. Final results for feed-forward neural networks in each group

| Group | Network | Structure | Training data | Performance | Test data | Performance |
|---|---|---|---|---|---|---|
| Group 1 | nnet11 | [7,18,3] | 701 | 91.44% | 707 | 80.10% |
| | nnet12 | [7,20,3] | 701 | 90.45% | 707 | 77.93% |
| Group2 | nnet21 | [7,4,4] | 560 | 87.50% | 187 | 90.37% |
| | nnet22 | [7,5,4] | 560 | 88.75% | 187 | 93.05% |
| | nnet23 | [7,6,4] | 560 | 92.50% | 187 | 92.51% |
| | nnet24 | [7,7,4] | 560 | 93.93% | 187 | 88.37% |
| Group3 | nnet31 | [7,8,3] | 509 | 94.11% | 168 | 89.88% |
| | nnet32 | [7,9,3] | 509 | 95.48% | 168 | 86.90% |
| Group4 | nnet41 | [8,15,3] | 531 | 88.32% | 173 | 79.53% |
| | nnet42 | [8,17,3] | 531 | 91.90% | 173 | 81.50% |

Best performing networks are finally chosen for further testing with noisy beats. Training diagrams showing the reduction in MSE versus the number of epochs are illustrated in Fig. 4.8., for the chosen networks. The final choice is the best network for each group, in terms of every indicator mentioned.

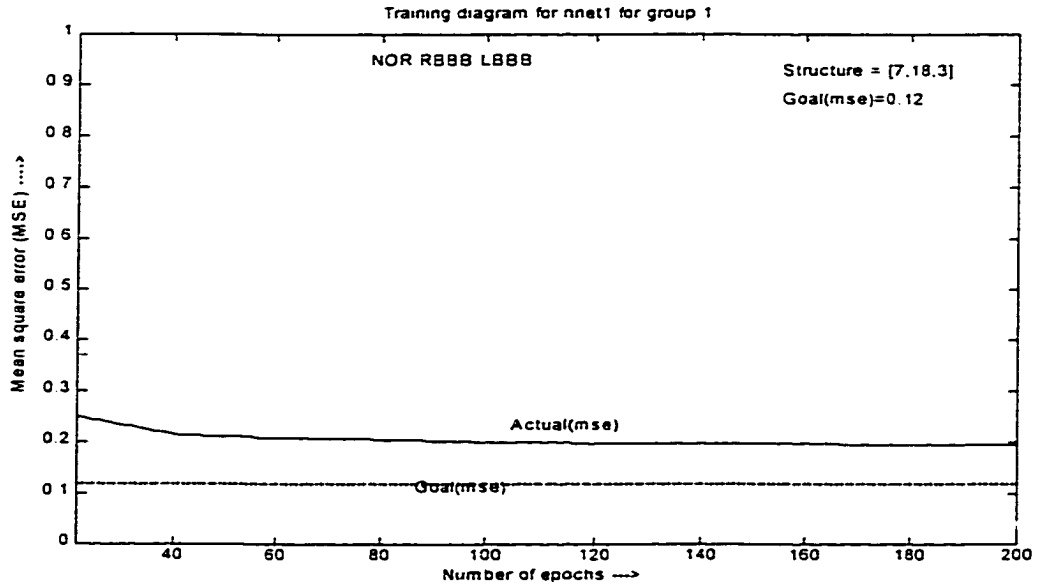



Fig. 4.11.(a). Training diagram for nnet1 for group 1'

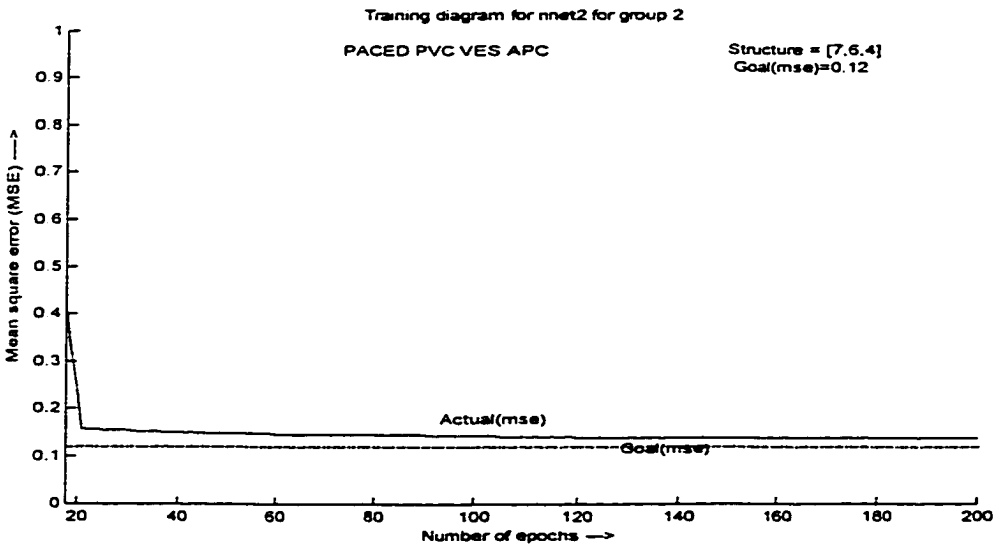



Fig. 4.11.(b). Training diagram for nnet2 for group 2

Fig. 4.11. Training diagram showing the reduction of MSE with training epochs

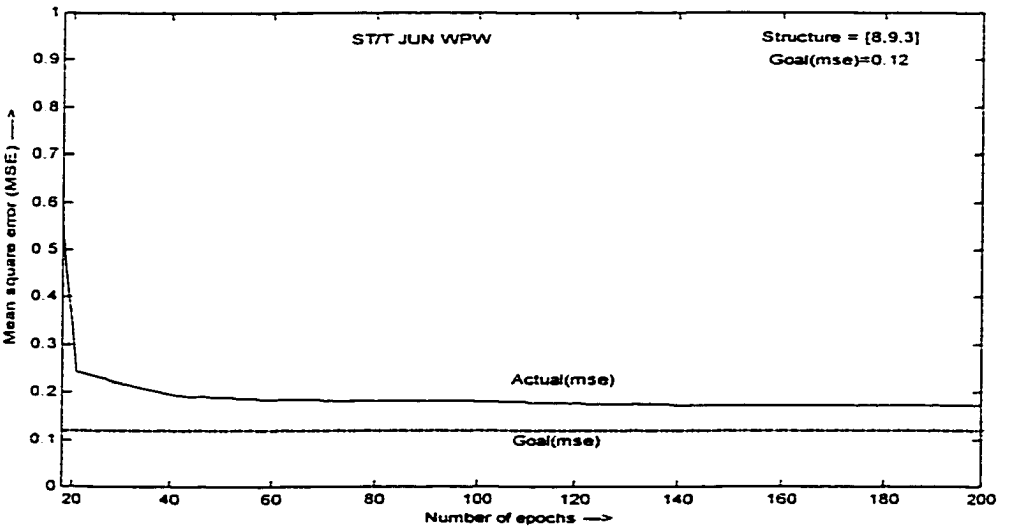



Fig. 4.11.(c). Training diagram for nnet3 for group 3

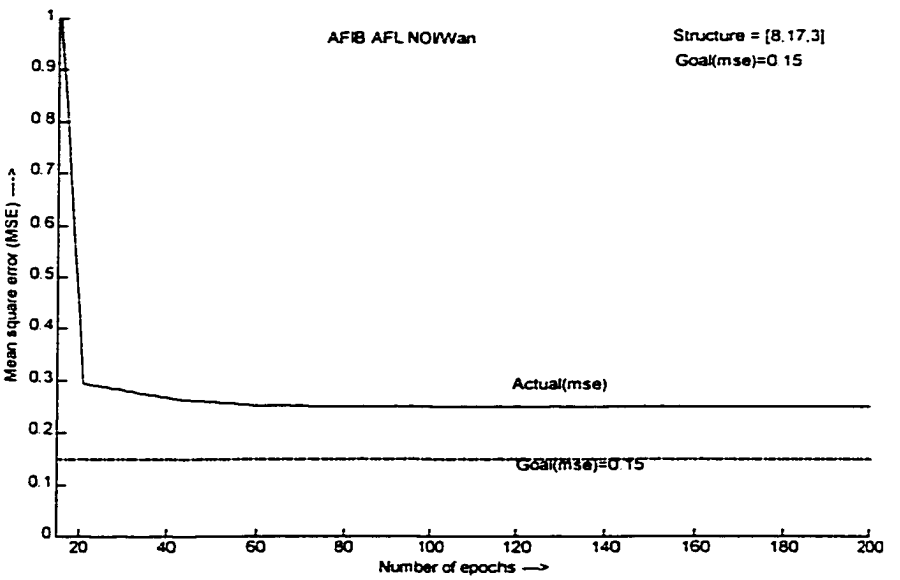



Fig. 4.11.(d). Training diagram for nnet4 for group 4

Fig. 4.11. Training diagram showing the reduction of MSE with training epochs (Cont.)

### 4.6.3. Testing with Noisy data

The methodology provides the feature extraction step as a buffer, eliminating the effects of non-stationarity of the raw signal and the noise characteristics. A total of 120 noise-free beats (10 for each class, other than noisy beats class) are chosen for the

analysis. Each one of the beats is added with additive gaussian noise in steps of 5% with respect to amplitude (2.5% with respect to power), after normalizing signal and noise amplitudes. Matlab-Simulink® tool box is used to generate the noisy beats. Performance of the fiducial point detection algorithm has already tested for such cases as presented earlier. Table.4.7 indicates the performance of beat classification neural network with increasing noise level. Performance of the algorithm with increasing noise power is compared with the performance of human experts on an undistorted data.

Table. 4.7. Results on testing with noisy data

| Additive noise power | Performance of fiducial points (Visual criteria) | Classification (Beat) |
|---|---|---|
| 2.5% | 98% | 114/120 |
| 3.6% | 96% | 109/120 |
| 4.9% | 93% | 105/120 |
| 6.4% | 91% | 99/120 |
| 8.1% | 87% | 96/120 |
| 10.0% | 85% | 93/120 |
| 12.1% | 83% | 90/120 |

4.6.4. Results of Rhythm Classification

Rhythm classification in the present work is viewed as a deterministic problem, since the rhythm classes essentially derive from heart rate class and the individual beat classes. A table showing the implementation is provided in chapter 3. The problem is similar to automatic character recognition in an LCD display. A hundred percent accuracy is obtained in this scheme. Hence, the accuracy of overall rhythm

classification in the present work is dependent on the individual beats. An unknown combination of input beats produce an output the same as the input beat classes. Such an approach enables the possibility of secondary opinion by human experts.

## 4.7. Summary of Results

All the algorithms are written as Matlab® functions. These functions are shown in Appendix E and comments for each function is shown. Any new user can apply the functions on appropriate inputs. The process of organization of data and features into an array structure enables easy implementation with neural networks. Providing the group number as input allows the user to select all the data belonging to a particular group. Any sub-class can also be selected using an input of five digits. Such a selection procedure increased the speed of implementation and enabled the evaluation of more than 200 neural networks. A complete set of conclusions derived from all the implementations performed is discussed in next chapter. The resulting program code contained almost 5,000 lines of MATLAB program code, requiring around 500 Kb of memory. Around 10-30 different neural networks can be trained in an 8 hour period using a 266 MHz Pentium computer. The exact number depends on number of input features, problem complexity, and the number of hidden neurons.

# CHAPTER 5

# DISCUSSION AND CONCLUSIONS

## 5.1. Overview

Computerized ECG interpretation algorithms incorporating methodologies to extract and use maximum information from the signal are needed for improving the reliability of the machines [Bessete and Nguyen, 1989; Garfein, 1990]. Computerization of such a vital clinical routine is bound to reduce costs, helps to overcome the unavailability of experts in rural clinics in emergency situations, and helps to increase consistency in diagnosis [Baxt, 1992]. Web-based diagnosis of routine ECGs may require consistent, quick responses for ECG waveforms acquired under varying machine specifications. Clinicians and cardiologists accept the ability of neural networks to improvise the performance in specific ECG related tasks [Devine and Macfarlane, 1993]. Neural network implementations have outperformed individual human experts and the first two generations of interpretation systems in:

(i) QRS wave and ST segment detection [Rasiah *et al.*, 1995; Suzuki and Ono, 1992; Xue *et al.*, 1992],

(ii) Detection of lead reversal in ECG recordings [Heden, 1996],

(iii) Relating clinical variables and ECG findings to location and size of infarction [Heden *et al.*, 1997; Baxt, 1992], and

(iv) Detection of cardiac hypertrophy [Ouyang *et al.*, 1999].

Technical and engineering research needs to provide a working schematic to solve more complex Type-A, B and C problems, providing expandable and easy-to-implement solutions using neural networks. This dissertation addresses the need by providing a bottom-up solution to the Type-C problem of classifying the common rhythm and morphological patterns [Trahanias and Skordalakis, 1989]. An essential concept in a bottom-up approach is that the perfection of individual sub-systems leads to improve overall system performance.

Type-C systems usually make use of the original classification by cardiologists and clinicians. Two different comparisons are provided for establishing the superiority in performance of the current approach. In the first comparison, an effort is made to compare the present results with a few results from the first two generations of ECG rhythm classification systems. Chapters 1 and 2 provided an overview of the first two-generation systems and the inherent limitations in them. The first comparison highlights this aspect as the reason for continuing distrust among cardiologists to rely on the machine decisions. The second comparison focuses on the specific area of neural-network based classification systems, especially in last few years. Both the comparisons are illustrated with tables.

The classification system presented here offers two major advantages compared to related research. The first advantage is the inclusion of fiducial point detection and feature extraction systems in implementation and the second one is the ability to include complementary information from frequency domain, a derivation of the first advantage. A third minor advantage may be the inclusion of noisy beats in implementation of every sub-system. Benefits of these three aspects are amplified through out this chapter.

Discussions focus more on the means, *i.e.*, methodology rather than on the ends, *i.e.*, results, which turn out to be among the best in comparison to results available in similar research.

## 5.2. Comments on Pre-processing

The pre-processing work having the maximum impact has been the re-sampling process, which expanded the data set. The re-sampling process made the whole system machine independent. Four broad types of re-sampling techniques were attempted, as discussed in section 4.1.2. FIR filters presented minor distortions nearer to the end-points, whereas the frequency domain techniques provided abnormal points compared to the original signal. Computations in digital domain, e.g., backward step interpolation and nearest neighbor interpolations, lack smooth appearance after re-sampling. The final choice is cubic interpolation, based on frequency domain parameters like maximum power frequency and power spectral density ratios and the smoother fit. Cubic interpolation provided a compromise between linear interpolation and spline interpolation based on simplicity and accuracy.

## 5.3. Comments on Fiducial Point Extraction

Fiducial point extraction and QRS detection algorithms form a major part of the present implementation. A high accuracy in both steps is the norm in last few years [Laguna *et al.*, 1994; Trahanias and Skordalakis, 1989]. A performance of nearly 99% is not uncommon in many new approaches utilizing wavelet transforms and neural networks *etc* [Li *et al.*, 1995]. Pan and Tompkins, 1986, achieved a near perfect performance in QRS detection and continuously improvised upon it [Xue *et al.*, 1992].

Most of these researches are performed with standard databases, especially MIT-BIH database. Accuracy of commercial algorithms has been quite good in the detection of QRS and fiducial point detection algorithms. There have been minor problems related to wave boundary estimations and the need for accuracy increases depending on the nature of information sought.

Interestingly, detection of fiducial points for beats belonging to fibrillation and flutter classes have not been attempted on a large scale nor discussed much statistically. Even manual definitions of exact fiducial points are difficult in such cases. Usually, those beats are detected early after QRS detection [Sornmo and Pahlm, 1984]. Discussions on the performance for noisy beats have been minimal.

In the present work, accuracy of the algorithms has been quite high for short duration signals. Noisy beats, paced beats, premature ventricular beats, atrial flutter and atrial fbrillation beats constitute around 30% of the data set in present work. Performance needed for fiducial point detection algorithms for such beats are kept minimal and two major criteria in accepting a beat for further classification tasks are:

(i) Ability to detect the QRS complex and to approximate the beat boundaries.

(ii) Ability to provide acceptable feature values in time domain.

Only three data segments out of 498 got rejected based on the above two criteria, though the overall rejection rate for beats is around 1.75%, at the input stage of the neural network. Inclusion of difficult classes and noisy beats seems to be the reason for this rejection rate. Multi-lead ECGs may play a significant role in further reduction, as the fiducial point detection is attempted in two or more channels, especially in cases of independent noise.

## 5.4. Comments on the Selection of Features and Groups of Classes

Extraction and selection of features has been vital as has been found during the implementations with out one [Kuppuraj, 1995; Tong, 1992; Lee, 1989]. Features reduce the problem space and provide an isolation of classification system from input noise and non-stationarity. The main reason assumed for the increased accuracy in present results seem to be the use of a feature set highlighting each group. A weak link in the present work is the absence of comprehensive theory on the selection of features. However, the advantage of easy implementation of neural networks enabled all possible combinations of features to be tested in each group as indicated in Appendix E. Such an approach seems to provide the neural network equivalent of regression analysis, and helps to determine the feature set with best possible training performance.

The selection of independent features for each group is another concern. In the present work, selection of features of each group is such that the outside beat classes produce completely different outputs for those features. This aspect provides another inherent statistical capability of training in neural networks [Schalkoff, 1992]. Feature sets are selected based on the standards suggested in literature and by regulatory authorities [Wagner, 1994; Rautaharju, 1998; Medtronic 9790 Programmer, 1999; Blackburn, 1999]. Still, better feature sets, in terms of accuracy and classification information they carry, are sure to help the implementation in a positive way.

## 5.5. Comments on Selection of the Network

The absence of a comprehensive theory on the choice of optimal structures for neural networks provokes a practical concept: selection of the optimal implementation

is determined by the optimal performance among various structures. Ease of implementation and minimal user interference required promote a quick comparison of various structures. Neural network implementations in ECG signal (or any other bio-signal) can be organized in two phases:

I Acquisition, selection and organization of the data and feature sets, and

II Training and testing of different structures.

Phase I typically consumes ten to twenty times more time compared with phase II. The Matlab® neural network toolbox made the phase II implementations considerably easier and provided the ability to test all possible structures in a feed-forward network. The selection of a particular network structure is determined by the ability to reach the training goal mean square error with minimal hidden neurons. An advantage of the neural network learning is that the outliers are progressively isolated in training compared to core feature space. Hence, mean square error does not directly indicate the number of errors in training and mean square error of 0.1-0.15 seems to be acceptable. Generalization to an unknown input is accurate with a mean square error of 0.10-0.15. Levenburg-Marquadt training provides an adaptive optimization in back-propagation and the number of training steps reduces to one-tenth to one-hundredth of traditional procedures like conjugate gradient and steepest descent algorithms with fixed step-size.

## 5.6. Evaluation of Classification Performance

Class-C system performance in ECG requires manual feedback for evaluation. Usually a group of experts provide a better gold standard compared to individual expert. The following two sections compare the accuracy of the present ANN-based system

utilizing bottom-up design principles with similar systems in first two generations and with other ANN systems presently developed.

### 5.6.1. Comparison with the First Two Generation Systems

Many of the early arrhythmia classification systems used single-lead information and especially lead II information and hence provide a valid comparison with the present system. Only a few of the well-known attempts are chosen for the present comparison. Leblanc, 1989, provided a detailed account of the problems associated with such systems. Consideration of noisy data is uniformly discarded in many cases and carefully chosen data with homogenous characteristics are used. Two major advantages of the present implementation over such systems are

(i) Increased accuracy of features in time and frequency domains.

(ii) Ease of training and testing a neural network, compared to statistical and syntactic tasks of similar complexity.

Table.5.1 provides a simple comparison of the types of rhythm classification attempted in the first two generations. Commercial products are assumed to provide a comparable performance, though detailed analysis of the errors are not forthcoming [Willems.J.J., 1990]. Two classes, namely fusion (FUS) beats and noisy (NOI) beats usually have two different annotations. For example, a noisy atrial premature (APC) beat provides two beats for training, one for APC class and another for NOI class. Similarly, a fusion involving a normal and a PVC beat provides two beats for training. Hence, noise is taken into account in training and constitutes around 6% of the total data set.

Comparisons of the various preprocessing strategies and network architectures used for ECG beat and rhythm related classification tasks are hampered by the size and generality of the data used to train and test the network, the difficulty of the classification task attempted and level of signal pre-processing employed. Based on both comparisons, the present system seems to be among the top few systems based on the number of classes, number of ECGs used for training and testing, and the final performance. Addition of noise indicators and re-sampling techniques provides an opportunity for developing into machine independent Type-B or Type-A analysis systems.

Individual relations of performance for particular beats are not analyzed. However, the major problem is in distinguishing the atrial premature beats (APCs) compared with normal beats. Group 2 always provides the best performance, due to relatively clear distinction in feature space, *i.e.* VES, PVC and PACED beats. Hence, APCs are added to the group 2. Another major problem is in distinguishing the ST and T wave morphologies from normal beats. Mismatch among groups occur for this classification.

Table. 5.1. Performance comparison with some selected systems in the first two generations

| Group,Year | Number of cases | Number of Classes | Details of Success | Implementation | Noise Consideration |
|---|---|---|---|---|---|
| Macfarlane and Lawrie, 1974 | 3200 | 40 rhythms. Beat classes -- non-specific | Sinus rhythm-97% Other rhythms-76% | Syntactic, decision tree approach. Beat classification unexplained. | None indicated |
| Klingeman and Pipberger, 1967 | 100 | Normal and ventricular hypertrophy | 60-80% | Multivariate Statistics | None indicated |
| Mozetic, 1990 (KARDIO) | In hundreds | 30 rhythms | 77% | Model-based, deep-knowledge, rule-based systems | None indicated |
| Van Bemmel *et al.*, 1990; Kors et al., 1996 [MEANS] | In thousands | Multiple rhythms | 80-95% | Comprehensive rule-based, systems. Features optimized. Extensive pre-processing. | Considered |
| Tong *et al.*, 1998 | In hundreds | Multiple rhythms | ≈80% | Knowledge-based | None indicated |
| Trahanias and Skordalakis, 1989 | 14292 QRS (125 subjects) | Selected rhythms | 80-90% 98%--feature extraction | Bottom-up approach to syntactic classification Feature extraction - extensive | Not considered |
| **Present System** | **3854** | **26 rhythms with unknown indications. 13 different beat classes** | **77-94% accuracy for different classes** | **Groups of feed-forward ANNs, with bottom-up approach** | **Included at every stage** |

Table.5.2. Comparison with more recent neural network implementations – not restricted to arrhythmia classification

| Group, Year | Number of Cases | Number of Classes | Details of Success | Implementation | Noise Consideration |
|---|---|---|---|---|---|
| Yang *et al.*, 1993 | In hundreds | Atrial fibrillation | 88.5%--92% | Feed forward NN<br>No feature extraction | None indicated |
| Lee *et al.*, 1989 | In hundreds | Ventricular tachycardia | 95% | Feed forward NN<br>No feature extraction | None indicated |
| Bortolen *et al.*, 1990 | 3266 patients[1] | Class-B/A systems[2] | 63% | Different architectures<br>No feature extraction | None indicated |
| Zong *et al.*, 1998 | 200 | Five beats and 20 rhythms | ≈70% | Fuzzy logic and neural networks | None indicated |
| Olmez *et al.*, 1998 | Minimal (in tens) | Seven beat types | ≈90% | Wavelet transform features and neural networks | None indicated |
| Minami *et al.*, 1999 | 175 | Ventricular tachycardia | ≈98% | Neural network on frequency domain features | None indicated |
| Ouyang *et al.*, 1999 | 78 | Hypertrophy | 95% | Feed-forward ANN with time domain features | None indicated |
| Bousseljot *et al.*, 1998 | 249 | 13 infarction types | 67.9% | Feed-forward ANN | None indicated |
| Xue *et al.*, 1998 | 1161 | 3 infarction types | ≈60-70% | Feed-forward (adaptive) | None indicated |
| **Present system** | **3854** | **26 rhythms with unknown indications. 13 beat classes** | **77-94% accuracy for beat classes** | **Groups of feed-forward ANNs, with bottom-up approach** | **Included at every stage** |

[1]Biggest study so far
[2] Infarction, hypertrophy

## 5.5. Future Directions

Single lead ECG beat and rhythm classification constitutes a vital step towards the implementation of Type-B and Type-A systems. Any clinically useful ECG interpretation system in a modern clinic employs the multi-lead ECG data. Speed and superior performance provided by neural networks make the online multi-lead applications possible. A specific sample showing the extension to multi-lead Type-A system has been illustrated. Development of web-based Type-B and Type-C systems would be the extension of the work presented in this dissertation.

### 5.5.1. Extending into a Multi-Lead Problem: a Case-Study

Many of the existing clinical ECG interpretation systems use rule-based syntactic approach. A sample application to detect cardiac infarction injury score (CIIS) is presented here. Original algorithm is derived from the Medical Algorithms project web-site [Svirbely and Sriram, 2000]. The algorithm may also have prognostic significance in predicting mortality rate due to coronary heart disease. An entire bottom-up implementation using a set of feed-forward neural networks is suggested in Fig.5.1.

Sub-system level implementation takes care of feature extraction in individual leads. Features suggested include duration of Q wave, amplitude of positive and negative T, amplitude of positive and negative QRS, ratios of Q/R ratio, *etc.* Such features are easily derivable in the present methodology and information is extracted from leads aVL, aVF, aVR, III, V1, V2, V3 and V5. In the proposed approach based on bottom-up design, a single neural network is chosen for each lead. Instead of discrete

values for contribution, an output linear layer can produce linear outputs. The outputs of the neural networks can be added to produce a score (CIIS).

Neural networks can be designed to produce a score depending on the feature values in each of the nine leads used. The sum of the scores indicates the final CIIS.

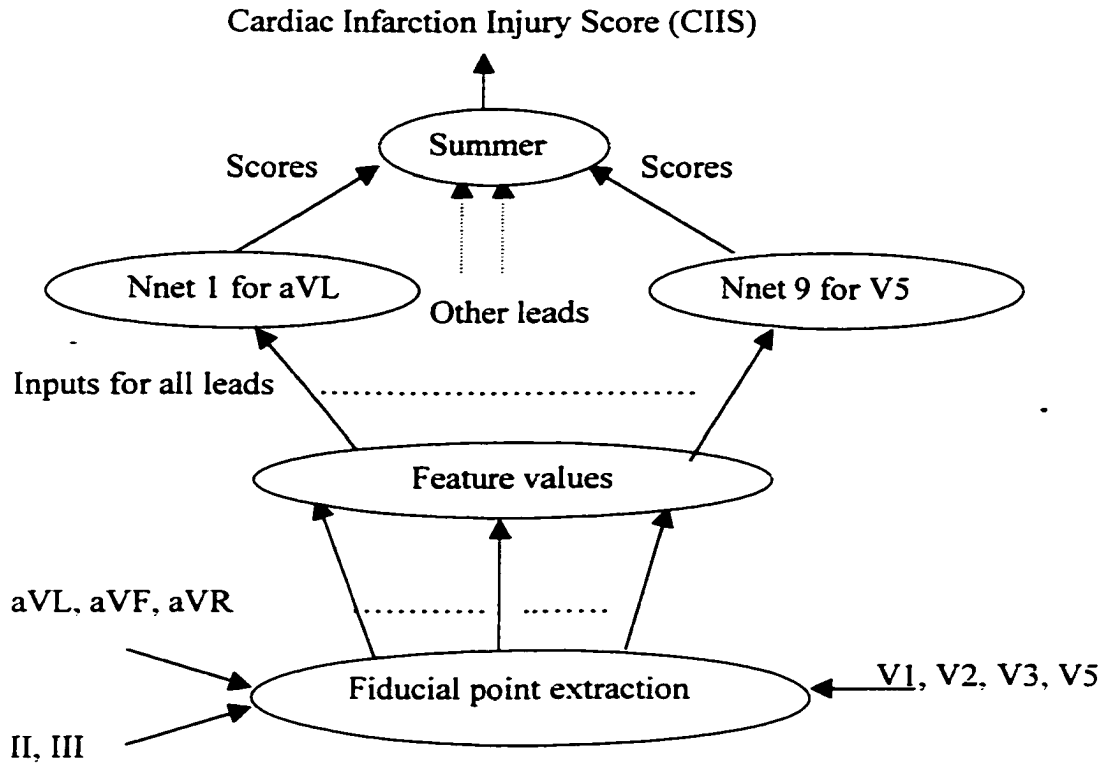



Fig.5.1. Nine separate small networks act on the information from nine channels

5.5.2. Other Directions

A bottom-up design to pattern classification using artificial neural networks is expandable to other signals also. Signals like electroencephalogram (EEG) from scalp electrodes and electromyogram (EMG) have been studied in frequency domain and features can be extracted to study multi-channel patterns. A theoretical question that

can be studied is the one related to the choice of features. With a fast implementation, choice of features can be identified iteratively first and a theory can be developed later. Another development can be the integration of clinical data along with ECG data. It would be interesting to see the influence of categorical variables on neural network design.

## 5.6. Summary

Based on above two comparisons, a clear case exists for further expanding the approach to typical multi-lead problems in distinguishing size and location of myocardial infarction, location of hypertrophy, *etc.* Type-A and B systems are easy to develop from present systems, as the syntactic, rule-based systems provide the essential information on the choice of features, choice of leads and their inter-relation. Groups of dependent neural networks for each lead are an easy way to visualize the solution. The present bottom-up approach provides an extendable approach in terms of time of recording, number of leads and problem space.

# APPENDIX A

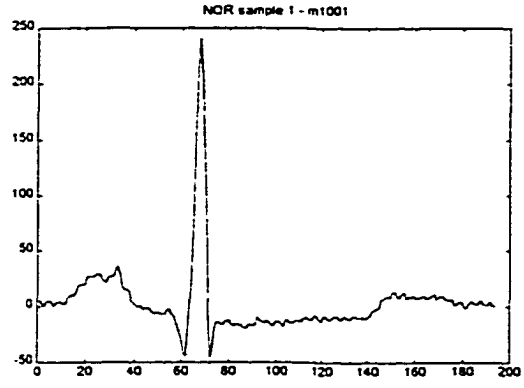
# SAMPLES OF BEAT CLASSES IN LEAD II

# APPENDIX A

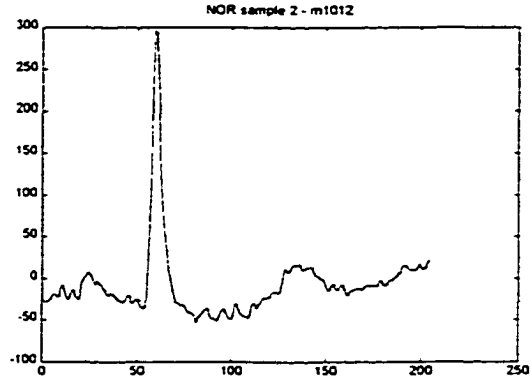
## SAMPLES OF BEAT CLASSES IN LEAD II

**Beat Class 1**: NOR – Normal beat

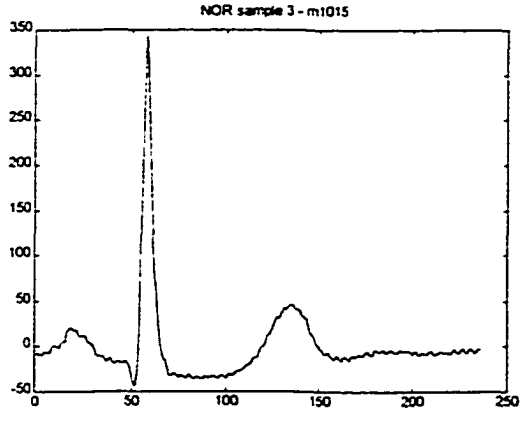
A normal beat in lead II usually has an upright QRS and T wave. Nine different morphologies are shown in Fig.A.1.(a)-(h). Variations are possible in the form of absence of definite S wave and in T wave shape. Duration of ST interval as well as PR intervals show variation. Polarity of the QRS complex is normally the same as that of the T wave. An iso-electric ST segment follows the QRS complex.



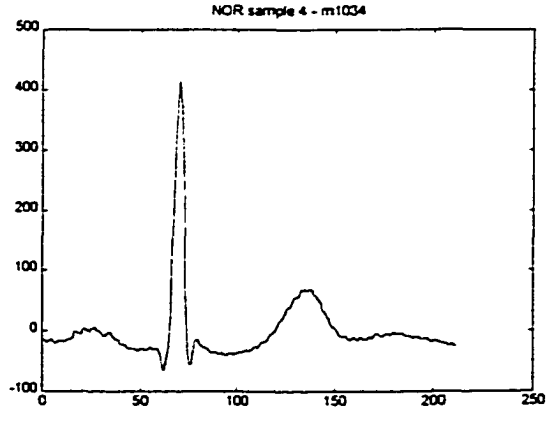

(a)




(b)




(c)




(d)

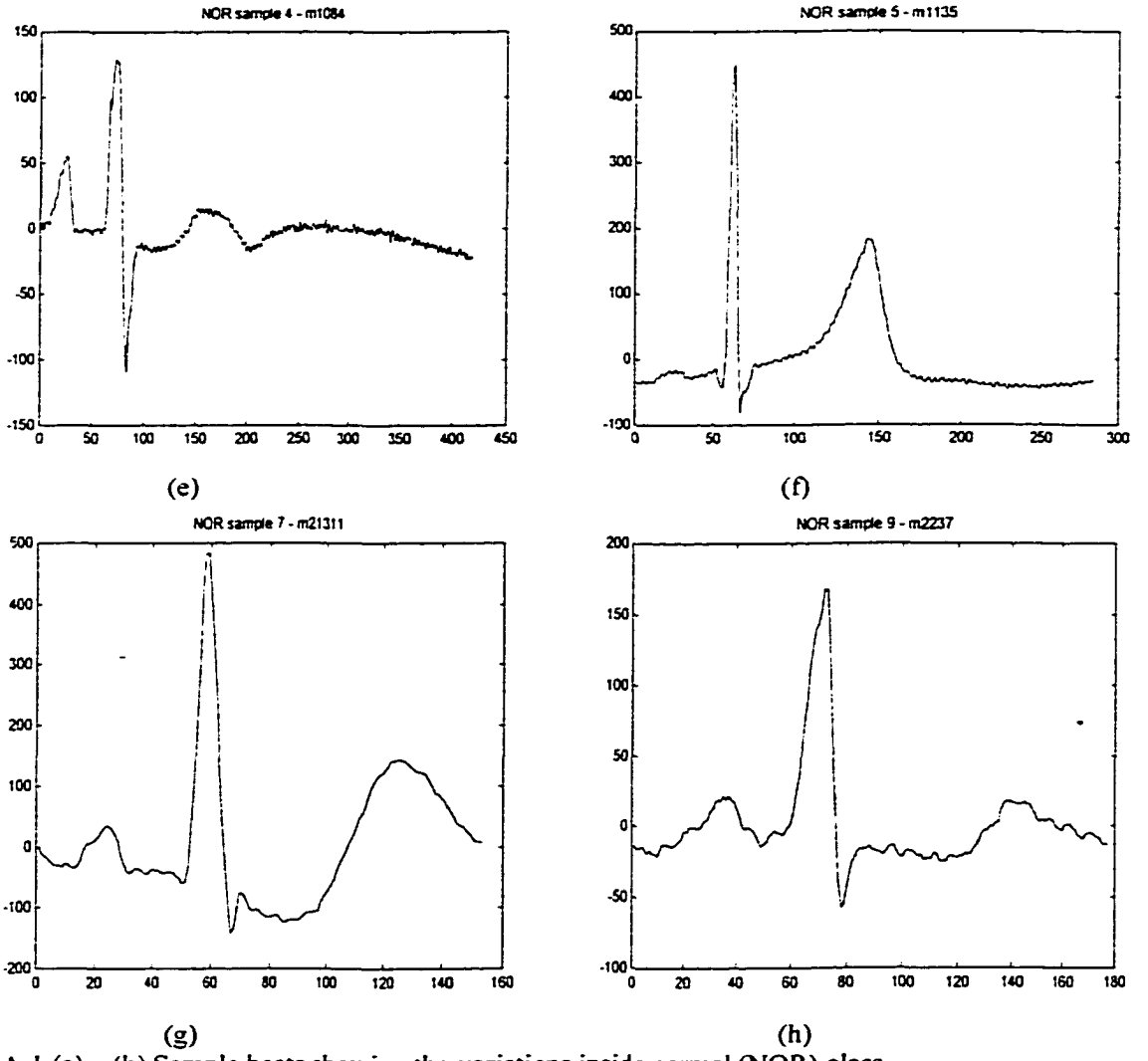



Fig.A.1.(a)—(h) Sample beats showing the variations inside normal (NOR) class

**Beat class 2**: RBBB – Right bundle branch block beat

The abnormal features of the ECG in right bundle branch block arise from delayed conduction of the wave depolarization over the right ventricle, which widens the QRS complex. Because depolarization is abnormal in BBB, so is repolarization. Thus the T wave is expected to be opposite in polarity to the terminal component of the QRS complex. Atrial activation is not affected.

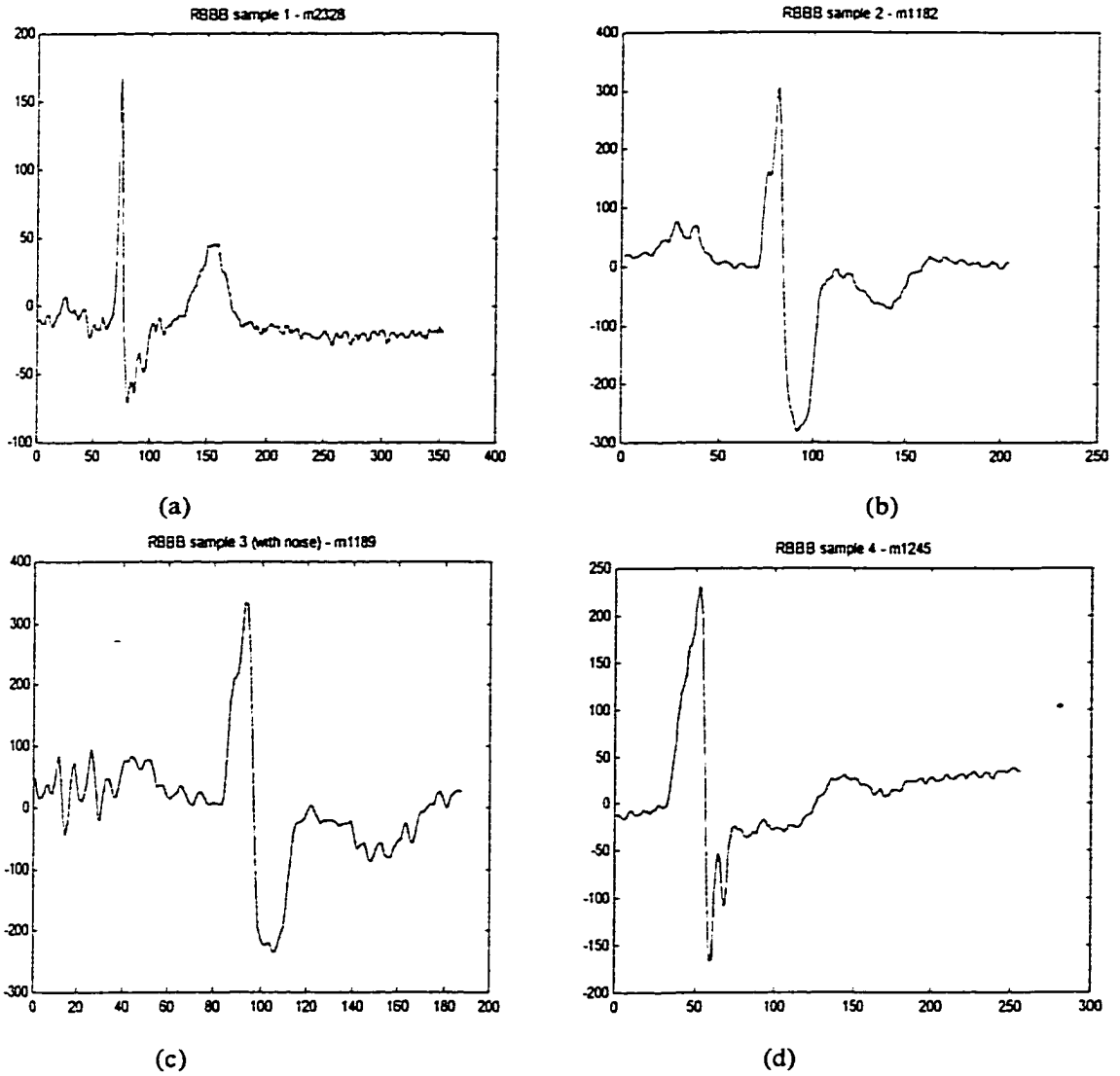



Fig.A.2.(a)--(d) Sample beats showing the variations inside RBBB class

**Beat class 3**: LBBB – Left bundle branch block beat

The ECG changes seen in LBBB beats include:

- Prolonged QRS duration (0.12 sec or more)
- Little or no q or S wave in lead II
- Direction of T wave is mostly opposite to that of QRS complex
- Usually, P wave characteristics remain unaffected.

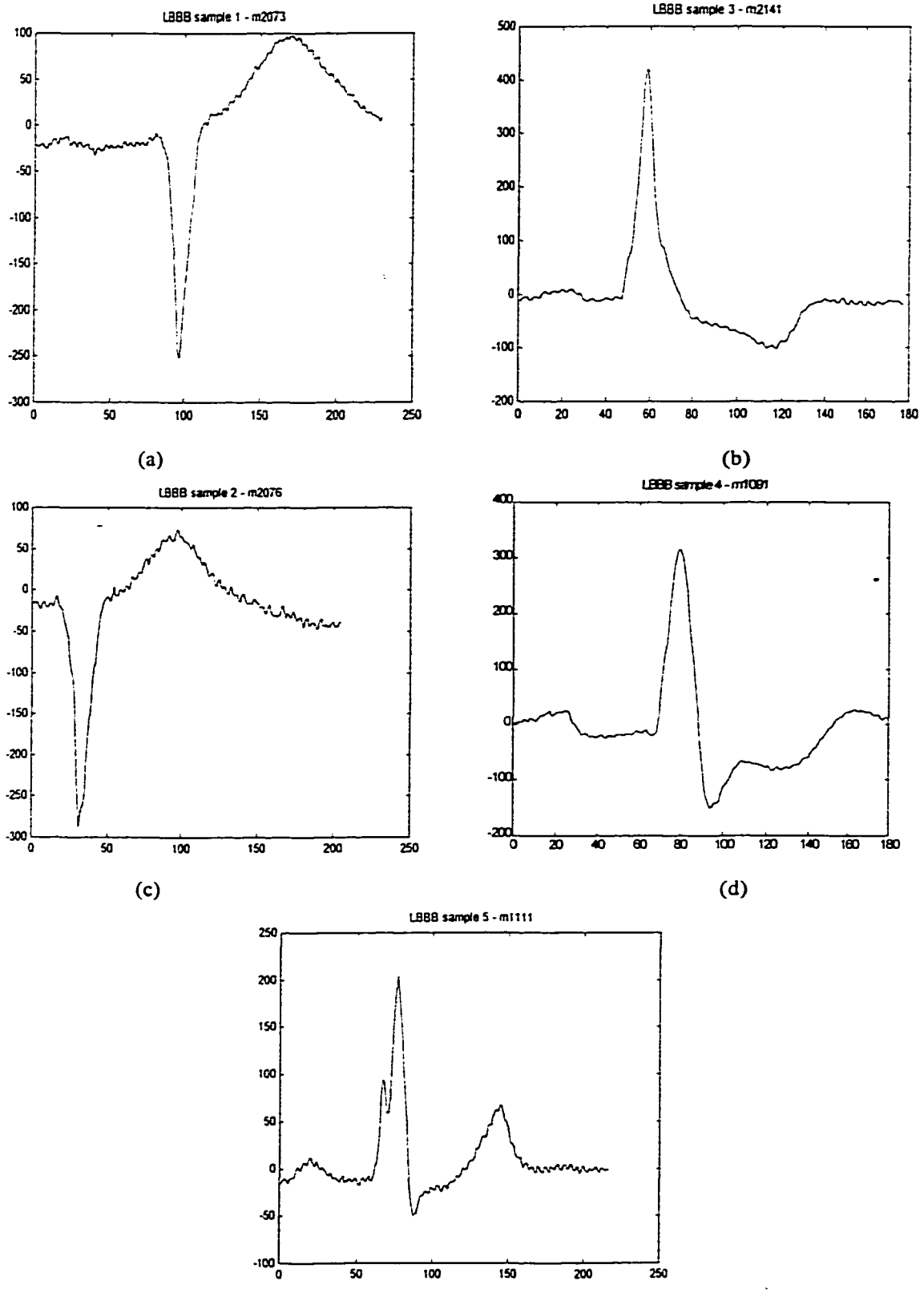
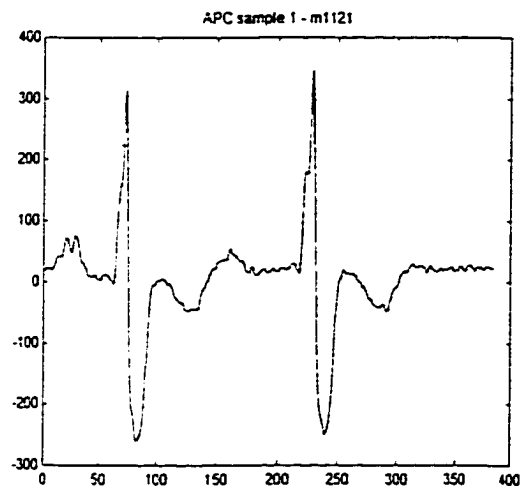


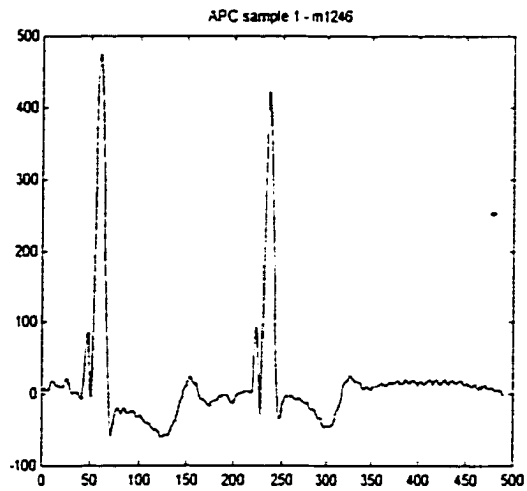

Fig.A.3.(a)-(e) Sample beats indicating the variations inside left bundle branch block (LBBB) class

**Beat Class 4**: Atrial premature contraction (APC) beat

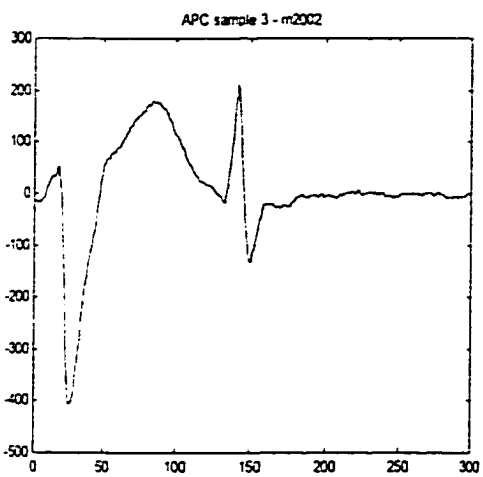
When an ectopic focus from the right or left atrium sends out an impulse before it is time for the next sinus node impulse, an early or premature contraction will occur. Indications are seen in the calulation of PP intervals and TP intervals in previous beats. Following the APC, a non-compensatory pause is usually seen on the ECG. The reason for the pause is the depolarization of SA node when the ectopic impulse travelled through atrium. The refractory SA node has to repolarize first, before accepting the next impulse from the vagus nerve. Features need to take into account the previous beat also. APCs may be multi-focal or unifocal.



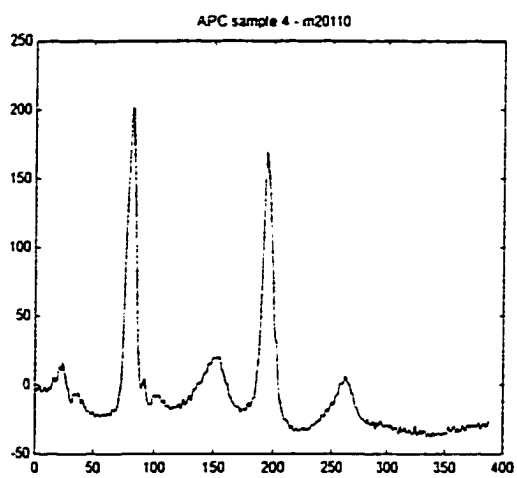

(a)



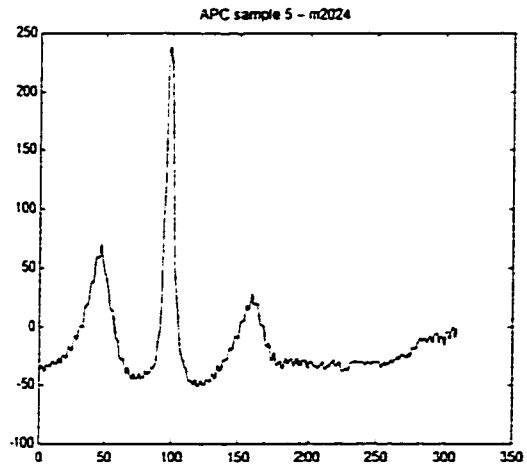

(b)




(c)




(d)

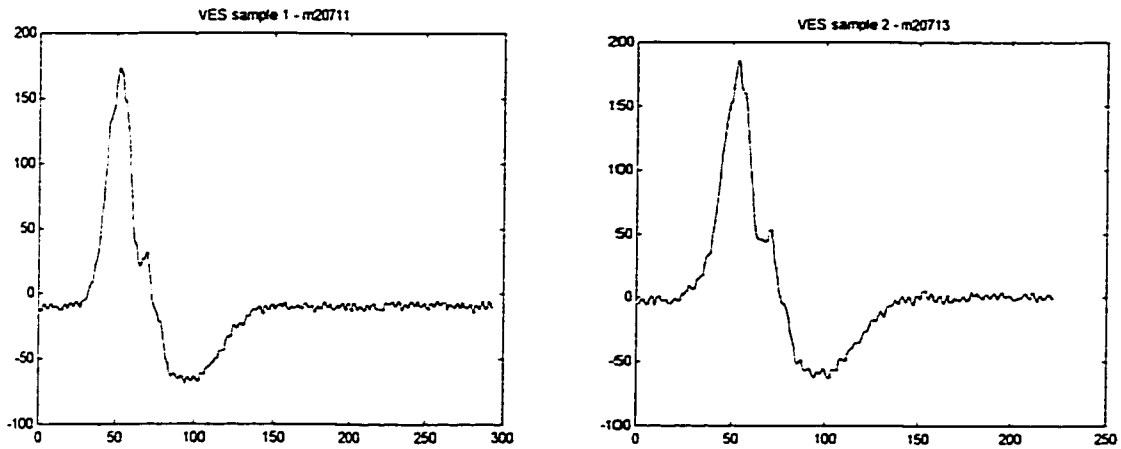



(e)

Fig. A.4.(a)—(e). Sample beats indicating the variations inside atrial premature contraction (APC) class

**Beat class 5**: Ventricular Escape beat

Independent of atrial activity with not much distortion of the QRS complex. Usualy the ventricular foci is same and the shape looks similar for any single subject.




(a) (b)

Fig. A.5.(a)—(b). Sample beats indicating the variations inside ventricular escape beat (VES) class

**Beat Class 6:** Premature ventricular contraction

PVCs are usually caused by ectopic impulses in atrial and junctional tissue. Usually, PVCs appear as wide and bizzarre looking QRS complexes. This appearance is due to the fact that the ventricular depolarization occurred along aberrant pathways. Usually no P wave preceds the QRS complex and R wave in opposite direction to the T wave and is followed by a full compensatory pause.

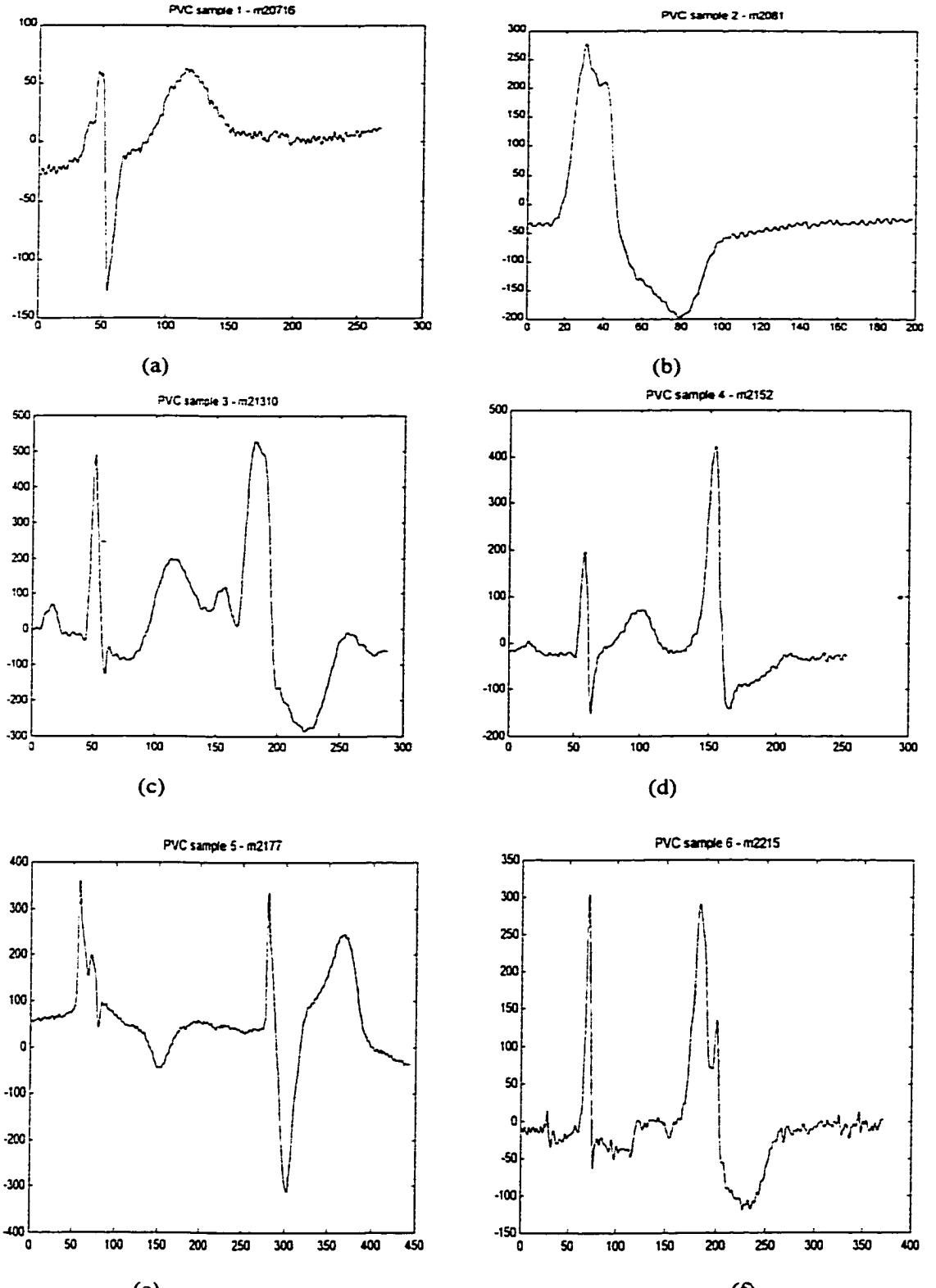
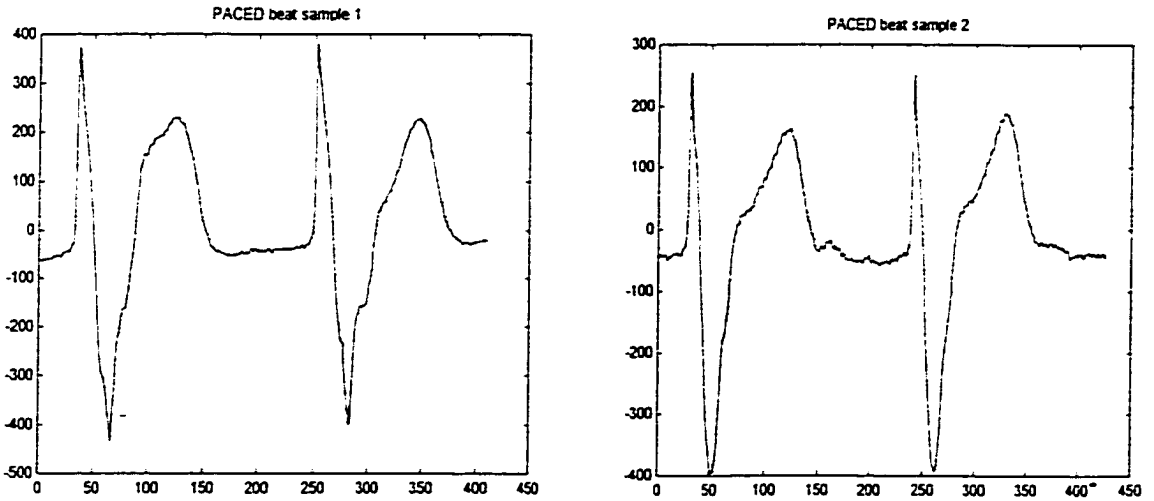



Fig.A.6.(a)--(f). Sample beats indicating the variations of premature ventricular contraction (PVC) class

**Beat Class 7:** Paced beats (PACE)

Paced beats are slightly different from the usual arrhythmia or beat classes in the sense that they are produced when the artificial pacemakers take over. They are uniform in shape and present a pattern matching ventricular activity.

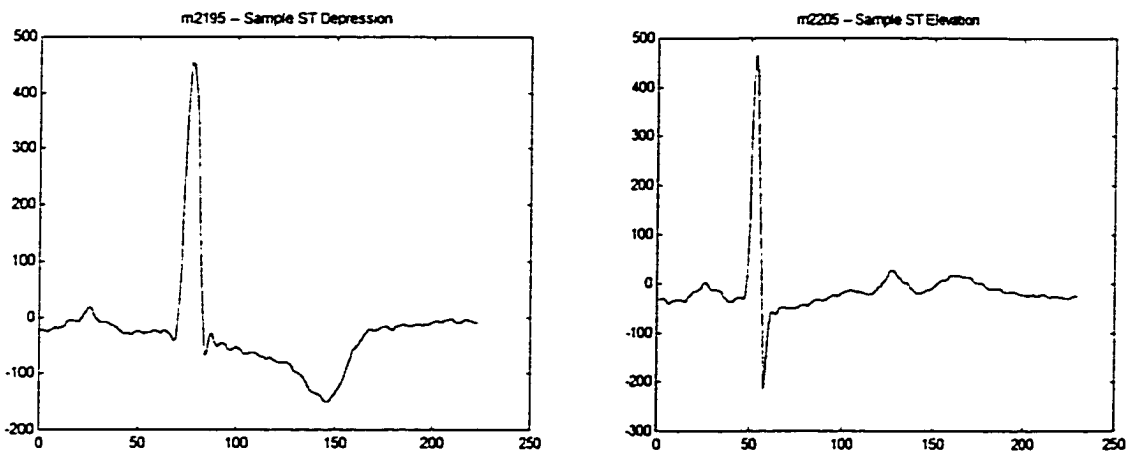



(a) (b)

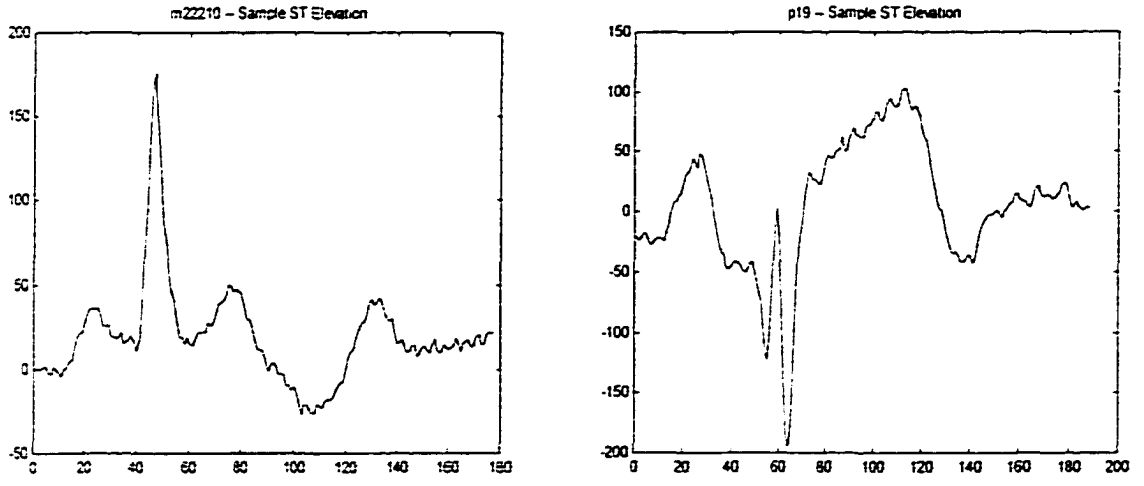
Fig. A.7.(a)—(b). Sample beats indicating the variations inside paced (PACE) class

**Beat class 8**: ST segment and T wave abnormalities

T wave inversions do not have any special significance but may occur together with ST segment depressions. ST segment disturbances occur for subjects during exercise or myocardial injury.




(a) (b)



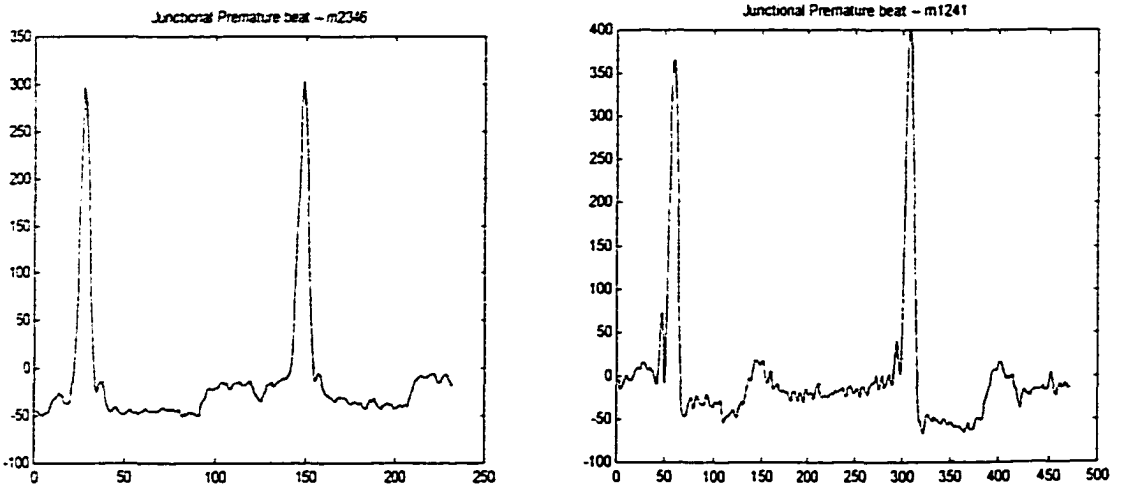

(b) (d)

Fig. A.8.(a)—(d). Sample beats indicating ST segment and T wave variations (ST-T) class

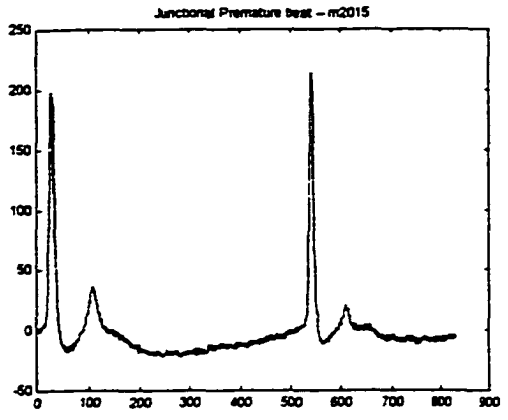
When the direction of ventricular repolarization gets affected, there is a good possibility of a visible change in pattern. These changes usually occur together with some effect on Q and S waves. P waves remain unafected.

**Beat class 9**: Junctional(JUN) Premature beat

Junctional (AV -nodal) beats and rhythms occur when the SA node or the atria fail to initiate impulse formation. The automaticity of the AV node can assume the pacemaker function and will pace the heart at an inherent rate of 50 to 70 beats/min. The beats occur for a period of time, before the SA node takes over. Conduction ratios are 1:1 or 0:1. P waves will occur before, or during or within QRS complex.




(a) (b)




(c)

Fig. A.9.(a)—(c). Sample beats indicating the variations inside junctional premature beat (NODAL) class

**Beat class 10**: Wolff-Parkinson-White Syndrome (WPW) (Preexcitation)

This is an uncommon congenital anomaly affecting conduction through the AV node. It appears that there is normal passage of SA node impulses through the AV node as well as accelerated passage of SA node impulses through pathways around the AV node. The result of this syndrome is an early activation of the ventricles, shortening of the AV node delay (P-R interval) and characteristic "R wave" fusing into the R wave. ST and T waves also get fused into QRS complex because of higher rate.

**Beat Class 11**: Atrial Flutter (AFL)

Ectopic atrial foci cause the atrial rate to be 250 or 350 beats/min. This produces the distinctive sawtooth pattern, because a P wave is produced with each atrial impulse. These waves are referred to as flutter (F) waves. The atrial rate is so fast that it does not allow this rapid transmission to the ventricles by the AV node. The AV node may conduct every 3$^{rd}$, 4$^{th}$, 5$^{th}$ or 6$^{th}$ atrial impulse. Atrial rates may go up to 250/300 per minute. PP interval remain usually same. ST and T waves fused with F waves. PR interval is difficult to measure.

**Beat Class 12**: Atrial Fibrillation (AFIB)

Exact etiology is not known but it is thought that that numerous ectopic foci transmit impulses so rapidly that the individual myocardial fibres of the atria depolarize and repolarize at independent

intervals. This causes erratic quivering of the atria. Atrial activity is uncountable and ventricular activity variable. ST and T waves are unidentifiable and P waves are erratic.

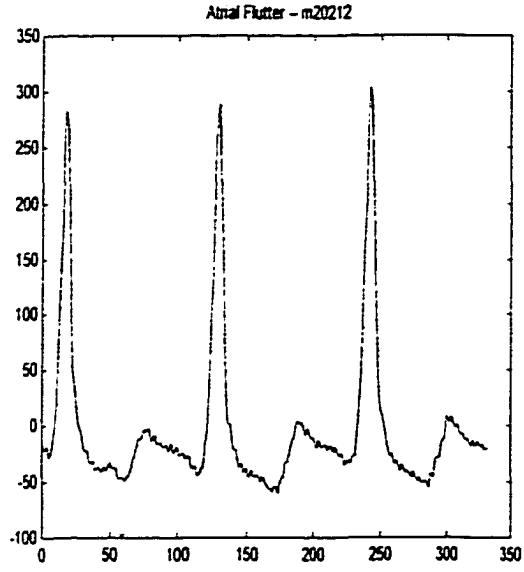



Fig. A.10. Sample atrial flutter beat sequence (AFL)

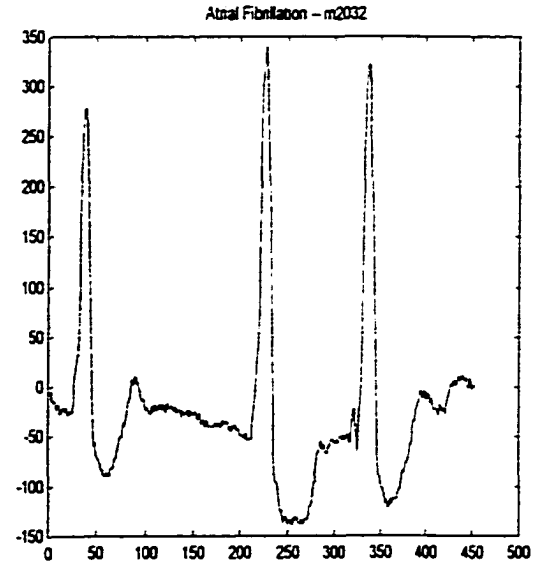



Fig.A.11. Sample atrial fibrillation sequence (AFIB)

# APPENDIX B

# ORGANIZATION OF ECG DATA

# APPENDIX B

## ORGANIZATION OF ECG DATA

Table. B.1.Data extracted from MIT-BIH database

| Data | Statement | Lead | Start | End |
|---|---|---|---|---|
| m1001 | NSR | MLII | 11:03 | 11:09 |
| m1011 | NSR | MLII | 1:38 | 1:44 |
| m1032 | APC | MLII | 19:19 | 19:25 |
| m1065 | VBIG | MLII | 12:27 | 12:33 |
| m1074 | PVC | MLII | 25:53 | 25:59 |
| m1097 | LBBB | MLII | 0:13 | 0:19 |
| m1249 | PVC | MLII | 27:42 | 27:48 |
| m2012 | AFIB | MLII | 0:07 | 0:13 |
| m2016 | AFIB | MLII | 7:55 | 7:61 |
| m2054 | PVC | MLII | 6:01 | 6:07 |
| m20711 | VESCAPE | MLII | 27:38 | 27:44 |
| m2125 | RBBB | MLII | 4:16 | 4:22 |

Table. B.2.Sample organization for data derived from MIT-BIH database

| No. | Name | Beat Classes |
|---|---|---|
| 1 | M1001 | NOR-NOR-NOR-NOR-NOR-NOR-NOR |
| 5 | M1011 | NOR-NOR-NOR-NOR-NOR-NOR(Wan) |
| 11 | M1032 | NOR-NOR-APC-NOR-NOR |
| 28 | M1065 | NOR-PVC-NOR-PVC-NOR-PVC-NOR-PVC |
| 37 | M1074 | PACED-PACED-PACED-PACED-PVC-PACED |
| 60 | M1097 | LBBB-LBBB-LBBB-LBBB-LBBB-LBBB-PVC-LBBB |
| 130 | M1249 | RBBB-RBBB-PVC-JUN-JUN |
| 155 | M2012 | AFIB-AFIB-AFIB-AFIB-AFIB-AFIB |
| 159 | M2016 | NOR-VCOUP-AFIB |
| 214 | M2054 | NOR-NOR-NOR-PVC-NOR-NOR-NOR-NOR |
| 233 | M20711 | VES-VES-VES-VES-VES |
| 276 | M2125 | RBBB-RBBB-RBBB-RBBB-RBBB-RBBB-RBBB-RBBB |

Table. B.3.Coding of classes for above segments

| No. | Name | Group & Class Codes |
|---|---|---|
| 1 | M1001 | 11000-11000-11000-11000-11000-11000 |
| 5 | M1011 | 11000-11000-11000-11000-11000 |
| 11 | M1032 | 11000-11000-10001-11000-11000-11000 |
| 28 | M1065 | 11000-2010-11000-2010-11000-2010-11000 |
| 37 | M1074 | 2001-2001-2001-2001-2010-2001 |
| 60 | M1097 | 10010-10010-10010-10010-10010-10010-2010-10010 |
| 130 | M1249 | 10100-10100-2010-3001-3001 |
| 155 | M2012 | 40100-40100-40100-40100-40100-40100-40100 |
| 159 | M2016 | 11000-2010-2010-40100-2010 |
| 214 | M2054 | 11000-11000-11000-2010-11000-11000-11000 |
| 233 | M20711 | 2100-2100-2100-2100 |
| 276 | M2125 | 10100-10100-10100-10100-10100-10100-10100-10100 |

# APPENDIX C

# STANDARD NOTATIONS USED FOR BEAT AND RHYTHM CLASSIFICATION

# APPENDIX C

## STANDARD NOTATIONS USED FOR BEAT AND RHYTHM CLASSIFICATION

Table. C.1. Notations for beat classes

| MIT-BIH Notation | Class Notation | Description |
|---|---|---|
| • | NOR | Normal beat |
| L | LBBB | Left bundle branch block beat |
| R | RBBB | Right bundle branch block beat |
| A | APC | Atrial premature beat |
| a | AAPC | Aberrated atrial premature beat |
| J | JUN | Nodal (junctional) premature beat |
| V | PVC | Premature ventricular contraction |
| F | FUS | Fusion of ventricular and normal beat |
| AFL | AFL | Atrial flutter beat |
| AFIB | AFIB | Atrial fibrillation beat |
| E | VES | Ventricular escape beat |
| P | PACE | Paced beat |
| Q | UNC | Unclassifiable beat |
| — | ST/T | ST elevation or depression and inverted Twaves |
| q/n | NOI/WAN | Noise and baseline wander |

Table. C.2. Notations for rhythm classes

| MIT-BIH Notation | Class Notation | Description |
|---|---|---|
| AB | ABIG | Atrial bigeminy |
| AFIB | AFIB | Atrial fibrillation |
| AFL | AFL | Atrial flutter |
| B | VBIG | Ventricular bigeminy |
| IVR | IVR | Idioventricular rhythm |
| N | NSR | Normal sinus rhythm |
| NOD | NOD | Nodal (A-V junctional) rhythm |
| P | PACE | Paced rhythm |
| PREX | PREX | Pre-excitation |
| SBR | SBR | Sinus bradycardia |
| SVTA | SVTA | Supraventricular tachyarrhythmia |
| T | VTRI | Ventricular trigeminy |
| VFL | VFL | Ventricular flutter |
| VT | VT | Ventricular tachycardia |

# APPENDIX D

# MATLAB ALGORITHMS DEVELOPED

# APPENDIX D

# MATLAB ALGORITHMS DEVELOPED

```
%Neural network related functions
function [x,y]=conjugate(initial,Coeff1,Coeff2)
% function [x,y]=conjugate(initial,Coeff1,Coeff2)
% This function tries to reach the optimized point using Conjugate gradient method.
% Inputs are
% initial=Initial value of the points [x(1);x(2)]
% matrix A= Hessian matrix for the function(not needed here)
% Coeff1=Coefficient of defF(x)/delx(1) first row [c11*x1+c12*x2]
% Coeff2=Coefficient of defF(x)/delx(2) second row [c21*x1+c22*x2]
% Developed by T. Srikanth 5/9/2000
%---------------------------------------------------------------------
function [x,y]=steep1(initial,Coeff1,Coeff2,alpha)
% function [x,y]=steep1(initial,Coeff1,Coeff2,alpha)
% This function tries to reach the optimized point using Steepest descent method.
% Inputs are
% initial=initial value of the points [x(1);x(2)]
% matrix A= hessian matrix for the function (not needed for Steepest-descent algorithms)
% Coeff1=Coefficient of defF(x)/delx(1) first row [c11*x1+c12*x2]
% Coeff2=Coefficient of defF(x)/delx(2) second row [c21*x1+c22*x2]
% alpha=learning rate
% Developed by T. Srikanth 5/9/2000
%---------------------------------------------------------------------
function c=rhythm(x)
% function c=rhythm(x)
% Relates the patterns in array x to one of the sixteen pattern classes and output c is the
% closest match.
% Each pattern contains 7 four-bit combinations.
% Developed bt T. Srikanth 5/9/2000
%---------------------------------------------------------------------
function [x,y]=steepany(initial, Coeff1, Coeff2)
% function [x,y]=steepany(initial, Coeff1, Coeff2)
% This function tries to reach the optimized point using Steepest descent method.
% The minimization is along each line.
% Inputs are
% initial=initial value of the points [x(1);x(2)]
% matrix A= hessian matrix for the function
% Coeff1=Coefficient of defF(x)/delx(1) first row [c11*x1+c12*x2]
% Coeff2=Coefficient of defF(x)/delx(2) second row [c21*x1+c22*x2]
% Developed by T. Srikanth 5/9/2000
%---------------------------------------------------------------------
function y=simp(x)
% This function calculates the simpson's integrator rule.
% Developed by T. Srikanth 5/9/2000
%---------------------------------------------------------------------
function [m1p,m2p,m3p,m4p,m5p,m6p]=param(f,m)
% This function estimates the five different power spectral density calculations based on the subtraction of P wave, QRS complex
and
% T waves. Two other power spectral density calculations are based on subtraction of all the waves and subtraction of a standard
ECG % wave corresponding to normal sinus rhythm of same PP interval as the observed beat. Last power spectral estimation
corresponds to % the original ECG wave.(beat-by-beat). Power spectral estimation is by Burg's estimation method. The inputs are
% f=index file corresponding to 9 fiducial points (2 for P wave, 2 for T wave, and 5 for QRS complex) for each beat
% m=data file with ecg beats corresponding to around 6 seconds duration and
```

```
% ECG signal is sampled at a rate of Fs=250 Hz.
% ECG data set m is normalized using a mean = 0  and standard deviation = 1;
% Some of the index files have the elements of last row to be zero. Only the first fiducial point from the last beat is needed. (for PP
% interval calculation).
% ECG data set is standardized with respect to amplitude in the range of mean=0 and variance=100;(adjustable);
% Subtraction is indicated in terms of base-line areas(TP interval mean).
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function [x,y]=newtonsany(initial)
% This function tries to reach the optimized point using Newtons descent method.
% Inputs are
% initial=Initial value of the points [x(1);x(2)]
% matrix A= Hessian matrix for the function
% Coeff1=Coefficient of defF(x)/delx(1) first row [c11*x1+c12*x2]
% Coeff2=Coefficient of defF(x)/delx(2) second row [c21*x1+c22*x2]
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function lse=leastsquares(e_in)
% This function calculates the least square value of any input vector e_in
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function [x,y]=newtons(initial, A, Coeff1, Coeff2)
% This function tries to reach the optimized point using Newtons descent method.
% Inputs are
% initial=Initial value of the points [x(1);x(2)]
% matrix A= Hessian matrix for the function
% Coeff1=Coefficient of defF(x)/delx(1) first row [c11*x1+c12*x2]
% Coeff2=Coefficient of defF(x)/delx(2) second row [c21*x1+c22*x2]
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function [pres_Wt,y,lms]=leastmeansquare(target,input,alpha,total_iterations, samples)
% This function tries to reach the optimized point using Least Mean Squares method.
% Otherwise called Widrow-Hoff rule. (LMS algorithm).
% Outputs are
% y = the complete weight changes and
% pres_Wt = the present value of the up-dated weights
% Inputs are
% initial = initial value of the points [x(1);x(2)]
% input   = input vector set (N * K) where N is the total number of samples and
% K is the number of features per sample.
% target  = target value of the points (N * M) where N is the number of outputs and
% M is the each output size [1] or [1;0] or [1;0;0] etc.
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function [x,y]=conjugate(initial,g)
% This function tries to reach the optimized point using Conjugate gradient method.
% Inputs are
% initial=Initial value of the points [x(1);x(2)]
% matrix A= Hessian matrix for the function(not needed here)
% Coeff1=Coefficient of defF(x)/delx(1) first row [c11*x1+c12*x2]
% Coeff2=Coefficient of defF(x)/delx(2) second row [c21*x1+c22*x2]
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function [x,y]=steepany(initial, A, Coeff1, Coeff2)
% This function tries to reach the optimized point using Steepest descent method.
% Inputs are
% initial-initial value of the points [x(1);x(2)]
% matrix A= hessian matrix for the function
% Coeff1=Coefficient of defF(x)/delx(1) first row [c11*x1+c12*x2]
% Coeff2=Coefficient of defF(x)/delx(2) second row [c21*x1+c22*x2]
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function count=compare(x,choice)
% function count=compare(x,choice)
% This function compares a particular column value with choice.
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function y=editvalue(x,choice)
% This function modifies the erraneous outputs using simple logical filters.
% Developed by Srikanth.T. on  May 3, 2000.
```

```
%-----------------------------------------------------------------------
function y=to_decimal(x)
% This function converts the binary digits to decimal integer.
% Developed by Srikanth.T. on April. 19. 2000.
%-----------------------------------------------------------------------
function s=select1(F1,targets,data_index)
% This function selects the features for required pattern recognition task.
% F1 = input from the function whole_ram
% nf = number of selected features.
% data_index=index of the data set.
% s(i,1) = local baseline amplitude
% s(i,2) = QRS shape and sign
% s(i,3) = T wave inversion(or) upright
% s(i,4) = TP interval
% s(i,5) = P1P2 interval
% s(i,6) = Noise reduction after smoothening
% s(i,7) = mean power frequency of the original beat
% s(i,8) = mean power frequency of the (original-simulated) beat
% s(i,9)-s(i,11) = power spectral densities.
% Developed by Srikanth.T. on March 18, 2000.
%-----------------------------------------------------------------------
function [overlap]=winsel(x,fs,n);
% This function basically selects the type of window using the various window options and calculates the windowed value of the
% original signal given the signal, the sampling frequency and the no. of pts for the window which should be same as the no. of
points
% for calculating the power spectral density. The function also has the overlapping and non-overlapping window option and sets the
% starting and ending point for the window according to the overlap specified during the course of the program. The windowed
value
% is calculated by just moving the window over the set of data points that represent the sampled signal. Then the various parameters
% are calculated for the windowed set of data. The syntax for the window option is
% winsel(signal,sampling frequency,no. of points of the window);
% Developed by T. Srikanth 5/9/2000
%-----------------------------------------------------------------------
function F1=whole_ram(ecg,Fs)
% This program calls other programs to calculate the features.
% This function calculates the features from the ECG beat in both time and
% frequency domains.
% F(1) =PP interval
% F(2) =mean base_line
% F(3) =QRS shape and sign (positive or negative).
% F(4) =ratio of areas under QRS and T waves.
% F(5) =JT1 interval
% F(6) =ST segment slope
% F(7) =PR interval
% F(8) =T wave inverted or upright
% F(9) =TP interval
% F(10)=Presence (or) absence of P wave (P1-P2) interval
% F(11)=ratio of upward Vs. downward slope for QRS complex.
% F(12)=ratio of R amplitude and Q amplitudes.
% F(13)=QT interval
% F(14)=Noise reduction after smoothening
% F(15)=Number of zero-crossings
% F(16)=TP interval(previous beat)
% Frequency domain features
% F(17:27)=frequency domain features
% F(17) = mean power frequency of the original beat
% F(18) = PSD ratio for the band (0.5-10) Hz original beat
% F(19) = PSD ratio for the band (10-20) Hz original beat
% F(20) = PSD ratio for the band (20-30) Hz original beat
% F(21) = mean power frequency of the original beat-simulated beat
% F(22) = mean power frequency of (0.5-10) Hz band
% F(23) = mean power frequency of (10-20) Hz band
% F(24) = mean power frequency of (20-30) Hz band
% F(25) = PSD ratio for the band (0.5-10) Hz
% F(26) = PSD ratio for the band (10-20) Hz
% F(27) = PSD ratio for the band (20-30) Hz
% Inputs are
% ecg = the particular beat and
% Fid = the set of fiducial points {P1,P,P2,Qs,Q,R,S,J,T1,T,T2,P1}
% pf1 = frequency domain parameters for that beat.
```

```
% Developed by Srikanth.T. on March 17, 2000.
%-------------------------------------------------------------------------
function [y1,y2,y3,y4]=srikpsdratio(psd,fs,b1,b2,b3,b4,b5)
% This function basically determines the psdratio of the given signal within the specified frequency band. Given the power spectral
% density points, sampling frequency,and the band of frequencies, the function first evaluates the fast fourier transform and then the
% total psd. Then it determines the psd ratio in the set of frequencies given and evaluates the psd ratio in each band, where the psd
% ratio is defined as the ratio between the sum of the psd in the given band to the sum of the entire psd.
% Syntax is [y1,y2,y3,y4]=srikpsdratio(psd,fs,b1,b2,b3,b4,b5)
% psd is the power spectral density estimate of the signal fs is the sampling frequency,b1,b2,b3,b4 is the frequency band for
% calculating the psd. Band is b1-b2,b2-b3,b3-b4,b4-b5,b5-b6;
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function s=selectall(F1,targets,data_index)
% This function selects all the features for required pattern recognition task.
% F1 = input from the function whole_ram
% nf = number of selected features.
% data_index=index of the data set.
% s(i,1:17) =  time-domain features and
% s(i,18:27) = frequency domain features
% s(i,28)=data_index
% s(i,29)=beat_index
% s(i,30:34)= targets ( 4 digit binary code)
% Developed by Srikanth.T. on April 1, 2000.
%-------------------------------------------------------------------------
function mxpf=srikmaxpowerfreq(psd,fs,f1,f2)
% This function is used to estimate the maximum power frequency of the given signal given the power spectral density and the
% sampling frequency. The maximum power frequency is defined as the frequencies at which the peaks occur in the power spectral
% density versus the frequency plot. First for the given sampling frequency, and the # of data points, a suitable frequency scale is
% obtained. Then the power spectral density points are taken and compared in order to find the number of peaks occurring. Then the
% frequencies corresponding to these peaks are estimated.
%The syntax for the function is y=srikmaxpowerfreq(power spectral density, sampling frequency,f1,f2);
% Modified on Mar 15, 2000.
%-------------------------------------------------------------------------
function s=select2(F1,targets,data_index)
% This function selects the features for required pattern  recognition task.
% F1 = input from the function whole_ram
% nf = number of selected features.
% data_index=index of the data set.
% s(i,1) = PP interval
% s(i,2) = ratio of areas under QRS and T waves
% s(i,3) = PR interval
% s(i,4) = TP interval
% s(i,5) = P1P2 interval
% s(i,6) = ratio of upward Vs. downward slope for QRS complex.
% s(i,7) = ratio of R Vs Q amplitudes
% s(i,8) = QT interval
% s(i,9) = Previous TP interval
% targets = 4 digit binary code for each beat (
% Developed by Srikanth.T. on April 1, 2000.
%-------------------------------------------------------------------------
function s=select3(F1,targets,data_index)
% This function selects the features for required pattern  recognition task.
% F1 = input from the function whole_ram
% nf = number of selected features.
% data_index=index of the data set.
% s(i,1) = PP interval
% s(i,2) = ratio of areas under QRS and T waves
% s(i,3) = PR interval
% s(i,4) = TP interval
% s(i,5) = P1P2 interval
% s(i,6) = ratio of upward Vs. downward slope for QRS complex.
% s(i,7) = ratio of R Vs Q amplitudes
% s(i,8) = QT interval
% s(i,9) = Previous TP interval
% s(i,10) = QRS shape and sign
% targets = 4 digit binary code for each beat
% Developed by Srikanth.T. on 5/9/2000.
%-------------------------------------------------------------------------
function n=zero_cross(ecg,local_base)
% This function calculates the number of times the local base-line is crossed.
```

```
% Developed by Srikanth.T. on April 1, 2000.
%-----------------------------------------------------------------------
function beat_type=detect(ECG)
% This is the final function which gives a call to all the other sub-functions. First step is the call to detect time-domain fiducial
points
% and subsequetly, a call to convert the original sampling frequency to 500 Hz. Next step, involves a call for time-domain features
%from 500 Hz signal. Frequency domain features are also simultaneously detected from the 500 Hz sampled signal. A call to select
% the features using BP network (time or frequency) is then followed by a call to detect the beat-class. This call will be making use
of
% one of the two alternatives, either time (or) frequency domain network.
% Developed by Srikanth.T. on 5/9/2000.
%-----------------------------------------------------------------------
function a=area(wave,local_base)
% This function calculates the area under T wave and QRS waves. Selects a portion from ECG signal and calculates the area.
% Developed by Srikanth.T. on Mar 14, 2000.
%-----------------------------------------------------------------------
function p=arpara(indexfile,m1p,m2p,m3p,m4p,m5p,m6p)
% This function estimates frequency domain parameters from the estimated power spectral estimations, m1p,m2p,m3p,m4p,m5p
and % m6p. Inputs are
% indexfile=index file containing the locations of fiducial points
% m1p=Power spectral calculations after the subtraction of P wave from the observed beat
% m2p=Power spectral calculations after the subtraction of QRS complex
% m3p=Power spectral calculations after the subtraction of T wave
% m4p=Power spectral calculations after the subtraction of all three components
% m5p=Power spectral calculations after the subtraction of a simulated ECG wave of
% similar characteristics as compared to the observed beat
% m6p=Power spectral calculations of the original observed beat
% parameters calculated include
% maximum power frequency and instantaneous frequency (overall), again in three bands of
% (0.5-10),(10-20) and (20-30) Hz ranges; Power spectral density ratio in these three
% bands is also calculated.
% total of 12 parameters for each beat and each power spectral estimation
% Syntax of functions called in this routine:
%  y=mpf(psd,fs,fstart,fend)
% [p1,q1,t1]=beat((L+d)/Fs,Fs,durat,ppeak,qpeak,tpeak,pt,qt,tt);
% [simul_wave,means,stds]=prestd(p1+q1+t1);
%  mxpf=srikmaxpowerfreq(psd,fs,f1,f2)
% [y1,y2,y3,y4]=srikpsdratio(psd,fs,b1,b2,b3,b4,b5)
% Developed by Srikanth.T. on Mar 14, 2000.
%-----------------------------------------------------------------------
function p=arpara_ram(indexfile,m1p,m2p)
% This function estimates frequency domain parameters from the estimated power spectral estimations, m1p,m2p. Inputs are
% indexfile=index file containing the locations of fiducial points
% m1p=Power spectral calculations of the original observed beat
% m2p=Power spectral calculations after the subtraction of a simulated ECG wave of similar characteristics as compared to the
% observed beat
% Parameters calculated include: instantaneous frequency (overall -- mean power frequency), again in three bands of (0.5-10), (10-
20)
% and (20-30) Hz ranges. Power spectral density ratios in three bands (0.5-6),(6-12) and (12-18) Hz are also calculated. Total of 12
% parameters for each beat and each power spectral estimation. Syntax of functions called in this routine:
%  y=mpf(psd,fs,fstart,fend)
%  [p1,q1,t1]=beat((L+d)/Fs,Fs,durat,pt,qt,tt);
%  [simul_wave,means,stds]=prestd(p1+q1+t1);
%  mxpf=srikmaxpowerfreq(psd,fs,f1,f2)
%  [y1,y2,y3,y4]=srikpsdratio(psd,fs,b1,b2,b3,b4,b5)
%  Last modified on April 25, 2000 by Srikanth.T.
%-----------------------------------------------------------------------
function [p,qrs,t]=beat(PP_inter,Fs,durat,ppeak,qpeak,tpeak,pt,qt,tt)
% This function generates the wave components of a ECG segment given the duration of PP intervals, Sampling frequency and the
% duration of recording.  PP intervals decide the pulse duty cycle of P wave generator. QRS interval is not disturbed unless the
% variation in PP interval is more than 0.10 sec. P wave and T waves are also less sensitive. But, the ST segment, PR segment and
TP
% segments vary in a ratio 1:2:8 for the changes in RR interval. Assumed heart rate is 75 beats per minute. (0.8 sec)
% duration = number of beats
% Input:
% PP_inter=beat interval in seconds
% durat=duration in number of beats for which the data is needed
% Fs   = Sampling frequency
% pt   = period of P wave
% tt   = period of T wave
```

```
% qt = period of QRS wave
% Output:
% p      = P wave sequence for number of beats selected
% qrs    = qrs complex sequence for number of beats selected
% t      = t wave sequence for number of beats selected
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function [p,qrs,t]=beat_ram(PP_inter,Fs,durat,pt,tt,qrst)
% This function generates the wave components of a ECG segment given the duration of PP intervals. Sampling frequency and the
% duration of recording. PP intervals decide the pulse duty cycle of P wave generator. QRS interval is not disturbed unless the
% variation in PP interval is more than 0.10 sec P wave and T waves are also less sensitive. But, the ST segment, PR segment and
TP
% segments vary in a ratio 1:2:8 for the changes in RR interval. Assumed heart rate is 75 beats per minute. (0.8 sec)
% duration = number of beats
% Input:
% PP_inter =beat interval in seconds
% durat=duration in number of beats for which the data is needed
% Fs=Sampling frequency
% pt=period of P wave
% tt=period of T wave
% qrst=period of QRS wave
% Output:
% p= P wave sequence for number of beats selected
% qrs= qrs complex sequence for number of beats selected
% t= t wave sequence for number of beats selected
% Modified on Mar 13, 2000.
%-------------------------------------------------------------------------
function [F1set,Error]=checkvalue(Fset,limit,columns)
% This function checks the feature values of the selected features in each beat. Allows the parameter values within 4 times the
% standard deviation.
% input is F, a matrix with multiple columns.
% output is F1, a new sub-set of features, which are free of outliers.
% Developed by Srikanth.T. on April 2, 2000.
%-------------------------------------------------------------------------
function count=classcount(dat,binary_code)
% This function calculates the number of times a particular class occurs, by matching the particular code. (five-digit code)
% Developed on April, 29, 2000.
%-------------------------------------------------------------------------
function [P1,P,P2,Qs,Q,R,S,J,T1,T,T2]=convert_fid(P1,P,P2,Qs,Q,R,S,J,T1,T,T2)
% This function adjusts fiducial points detected earlier with Fs=250 Hz into new points for new sampling frequency, Fs=500 Hz.
% Developed by Srikanth.T. on March 13, 2000.
%-------------------------------------------------------------------------
function F2set=doublecheck(Fset,N,limit)
% This function checks the feature values of the selected features in each beat. Allows the parameter values within 4 times the
% standard deviation.
% input is F, a matrix with multiple columns.
% output is F1, a new sub-set of features, which are free of outliers.
% input limit is an array with a length N:
% Developed by Srikanth.T. on April 2, 2000.
%-------------------------------------------------------------------------
function [data_feature,targets] = freqtime()
% freqtime: Distinction between time and frequency feature selection
%[data_feature,targets] = freqtime(), Returns:
%ALPHABET - 35x26 matrix of 5x7 bit maps for each letter.
%TARGETS  - 2x2 target vectors.
% Mark Beale, 1-31-92
% Revised 12-15-93, MB.
% Copyright (c) 1992-1998 by the MathWorks, Inc.
% $Revision: 1.6 $  $Date: 1997/12/30 19:39:30 $
%-------------------------------------------------------------------------
function [f,xp]=powsrik(x,fs,N)
% xpsd = spectra of the signal
% x = input signal
% fs = sampling interval
% N = Number of points of FFT
% f=frequency
%This function determines the power spectral density of the given signal, given the sampling frequency and the number of points
for
% fft. This is done by first estimating the length of the given signal which gives the no. of points for the fast fourier transform. Then
it
```

```
% calculates the fast fourier transform for the given signal and then estimates the power spectral density by multipying the fft value
% with its complex conjugate. The syntax for the function is
% [f,xp]=powram(signal,sampling frequency,no. of points for fft);
% Developed by Srikanth.T. on 5/9/2000.
%-----------------------------------------------------------------------
function y = parametric(P,Q,A,B,N)
% The program generates a complex AR, MA or ARMA time series given the filter parameters, excitation noise variance, and a
% complex array of zero mean, unit variance uncorrelated random variables.(Complex white noise). For an AR and ARMA process,
% the starting transient is eliminated since the initial conditions of the filter are specified to place the filter output in statistical
steady
% state. At present we can restrict everything to be real.
% y = output of the parametric model generator
% P = order of the AR part
% Q = order of the MA part
% A = AR Co-efficients in a array (Size = P)
% B = MA Co-efficients in a array (Size = Q+1)
% SIG2 = Variance of excitation noise
% N = number of data points desired
% V = Real white gaussian noise array of dimension (N+Q)*1
% Developed by Srikanth.T. on 5/9/2000.
%-----------------------------------------------------------------------
function [m1p,m2p]=param_ram(f,m)
% This function estimates two different power spectral density calculations. The first power spectral density calculation is done
after
%the subtraction of a standard ECG wave corresponding to normal sinus rhythm of same PP interval as the observed beat. Last
power
% spectral estimation corresponds to the original ECG wave (beat-by-beat). Power spectral estimation is by Burg's estimation
method.
% The inputs are
% f=index file corresponding to 9 fiducial points (2 for P wave, 2 for T wave, and 5 for
% QRS complex) for each beat
% m=data file with ecg beats corresponding to around 6 seconds duration and
% ECG signal is sampled at a rate of Fs=500 Hz.
% ECG data set m is normalized using a mean = 0 and standard deviation = 1;
% Some of the index files have the elements of last row to be zero.
% Only the first fiducial point from the last beat is needed (for PP interval calculation).
% ECG data set is standardized with respect to amplitude in the range of mean=0 and
% variance=100;(adjustable);
% Subtraction is indicated in terms of base-line areas(TP interval mean).
% Modified on March 14, 2000 by Srikanth.T.
%-----------------------------------------------------------------------
function [m1p,m2p,m3p,m4p,m5p,m6p]=param(f,m)
% This function estimates the five different power spectral density calculations based on the subtraction of P wave, QRS complex
and
% T waves. Two other power spectral density calculations are based on subtraction of all the waves and subtraction of a standard
ECG
% wave corresponding to normal sinus rhythm of same PP interval as the observed beat.
% Last power spectral estimation corresponds to the original ECG wave.(beat-by-beat).
% Power spectral estimation is by Burg's estimation method.
% The inputs are
% f=index file corresponding to 9 fiducial points (2 for P wave, 2 for T wave, and 5 for QRS complex) for each beat
% m=data file with ecg beats corresponding to around 6 seconds duration and
% ECG signal is sampled at a rate of Fs=250 Hz.
% ECG data set m is normalized using a mean = 0 and standard deviation = 1;
% Some of the index files have the elements of last row to be zero. Only the first fiducial point from the last beat is needed. (for PP
% interval calculation). ECG data set is standardized with respect to amplitude in the range of mean=0 and
variance=100;(adjustable);
% Subtraction is indicated in terms of base-line areas(TP interval mean).
% Developed by Srikanth.T. on 5/9/2000.
%-----------------------------------------------------------------------
function y=mpf(psd,fs,fstart,fend)
% This function basically determines the mean power frequency of any signal given the psd, and the sampling frequency,the starting
% and end frequencies.
% The syntax for the mean power frequency is y=mpf(psd,fs,fstart,fend)
% Developed by Srikanth.T. on Mar 16, 2000.
%-----------------------------------------------------------------------
function in=initselect(dat,col,checkno)
% This function identifies the proper set for each group.
% By Srikanth.T. on April 5, 2000.
%-----------------------------------------------------------------------
```

```
function ms=meanslope(wave)
% This function calculates the mean slope of the ST segment as well as upward and downward slopes of QRS complex.
% Developed by Srikanth.T. on March 14, 2000.
%-------------------------------------------------------------------------------
function in=initdiscard(dat,col,checkno)
% This function identifies the proper set for each data.
% By Srikanth.T. on April 5, 2000.
%-------------------------------------------------------------------------------
function m1=highsample(m,Fs)
% This function samples the original signal with a higher sampling rate of 500 Hz.
% m  = original signal
% Fs = 250 (or) 360 Hz
% m1 = output signal
% Developed by Srikanth.T. on Mar 13, 2000.
%-------------------------------------------------------------------------------
function s=select1(F1,targets,data_index)
% This function selects the features for required pattern recognition task.
% F1 = input from the function whole_ram
% nf = number of selected features.
% data_index=index of the data set.
% s(i,1) = local baseline amplitude
% s(i,2) = QRS shape and sign
% s(i,3) = T wave inversion(or) upright
% s(i,4) = TP interval
% s(i,5) = P1P2 interval
% s(i,6) = Noise reduction after smoothening
% s(i,7) = mean power frequency of the original beat
% s(i,8) = mean power frequency of the (original-simulated) beat
% s(i,9)-s(i,11) = power spectral densities.
% Developed by Srikanth.T. on March 18, 2000.
%-------------------------------------------------------------------------------
function [f,y]=aver2(navgs);
% Let us assume an averaging filter applied on an ECG signal. Let the signal sampling frequency be Fs=250 Hz and let the period
% be Np*Ts=0.8 sec. Hence, averaging is taking place over 200 points. Frequency response is also over 200 points. This file is in a
% script format , easy to take plot. Easy to genralize for any N
% Developed by Srikanth.T. on 5/9/2000.
%-------------------------------------------------------------------------------
function [f,y]=aver3(navgs)
% Let us assume an averaging filter applied on an ECG signal. Let the signal sampling frequency be Fs=250 Hz and let the period
% be Np*Ts=0.8 sec. Hence, averaging is taking place over 200 points. Frequency response is also over 200 points. This file is in a
% script format , easy to take plot.  Easy to generalize for any N
% Developed by Srikanth.T. on 5/9/2000.
%-------------------------------------------------------------------------------
function y=averfilt(N,Fs)
% This function calculates the amplitude response of the averaging filter based on the inputs shown below.
% y  = Output(i.e.) magnitude response of the averaging filter
% N  = Number of averaging intervals
% Fs  = Sampling frequency in Hertz
% Fixed input values
% Developed by Srikanth.T. on 5/9/2000.
%-------------------------------------------------------------------------------
function Pxx=avper(x,No_wind,N_fft,Wind_length)
% Pxx=avper(x,No_wind,N_fft,Wind_length)
% Developed by Srikanth.T. on 5/9/2000.
%-------------------------------------------------------------------------------
function [p,qrs,t]=beat(PP_inter,Fs,durat,pt,tt,qrst)
% This function generates the wave components of a ECG segment given the duration of PP intervals, Sampling frequency and the
% duration of recording. PP intervals decide the pulse duty cycle of P wave generator. QRS interval is not disturbed unless the
% variation in PP interval is more than 0.10 sec P wave and T waves are also less sensitive. But, the ST segment, PR segment and
TP
% segments vary in a ratio 1:2:8 for the changes in RR interval. Assumed heart rate is 75 beats per minute. (0.8 sec)
% duration = number of beats
% Input:
% PP_inter=beat interval in seconds
% durat=duration in number of beats for which the data is needed
% Fs  = Sampling frequency
% pt  = period of P wave
% tt  = period of T wave
% qrst = period of QRS wave
% Output:
```

```
% p= P wave sequence for number of beats selected
% qrs = qrs complex sequence for number of beats selected
% t= t wave sequence for number of beats selected
% Developed by Srikanth.T. on 5/9/2000.
%-------------------------------------------------------------------------
function y=pacf(x,N)
% This function calculates the partial autocorrelation based on Durbin Algorithm.
% x= input signal
% N= length of calculation for PACF function
% Developed by Srikanth.T. on 5/9/2000.
%-------------------------------------------------------------------------
function y = parametric(P,Q,A,B,N)
% The program generates a complex AR, MA or ARMA time series given the filter parameters, excitation noise variance, and a
% complex array of zero mean, unit variance uncorrelated random variables.(Complex white noise). For an AR and ARMA process,
% the starting transient is eliminated since the initial conditions of the filter are specified to place the filter output in statistical
steady
% state. At present we can restrict everything to be real.
% y = output of the parametric model generator
% P = order of the AR part
% Q = order of the MA part
% A = AR Co-efficients in a array (Size = P)
% B = MA Co-efficients in a array (Size = Q+1)
% SIG2 = Variance of excitation noise
% N = number of data points desired
% Developed by Srikanth.T. on 5/9/2000.
%-------------------------------------------------------------------------
function [f,xp]=powsrik(x,Fs,N)
% xp=output spectra of the signal
% x=input signal
% Fs=Sampling frequency in hertz
% N= Number of points of fourier transform
% f=frequency
% Developed by Srikanth.T. on 5/9/2000.
%-------------------------------------------------------------------------
function y4=qrsindex(yi,y3)
% This function calculates the start and end-points of QRS complexes in an ECG segment.
% yi=input signal (ECG) to moving window integrator
% y3=moving window integrator output
% y4=an array with calculated start and stop points
% Developed by Srikanth.T. on 5/9/2000.
%-------------------------------------------------------------------------
function [y,y1,y2,d,z2]=slowcurve2(dat)
% This function finds if the dat satisfies the conditions for a curve fit  for a region belonging to P or T region.
% Let ld=length of dat be an odd number;
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function y=smoothen(x)
% The function smoothens x, with a scheme of local averaging. Output signal y can be used for feature extraction.
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function [y,y1,y2,d]=slowcurve3(dat)
% This function finds if the dat satisfies the conditions for a curve fit  for a region belonging to P or T region.
% Let ld=length of dat be an odd number;
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function c=stline(x,factor1)
% This function checks if the signal x can be considered as a straight line.
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function c=straightline(x,r_max)
% This function checks if the signal x can be considered as a straight line.
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function c=testcurve(dat)
% This function tests if the given data segment is really a curve, based on the straight line and curve portions of a given data.
% In a data of L points, first L/2 points should satisfy a base-line and the rest should satisfy a curve. There are some exceptions to
this
% case.
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
```

```
function [m,new]=truebaseline(ecg,u)
% This function calculates the average baseline value of the ecg signal, after the removal of QRS complexes. Inputs are:
% ecg = ecg signal and
% u         = [Qs:J]
% Revised on 12/26/99 by Srikanth.T.
%-------------------------------------------------------------------------
function fid=turningpoint(x)
% This function utilizes the raw ecg and the single point derivative to calculate the P peak, QRS peaks and the T peak values.(any
% other peaks also).
% fid = fiducial point indices
% x   = input signal
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function [y_sq,y_st,y_ab]=slowcurve1(dat)
% This function finds if the data satisfies the conditions for curve fit for a region belonging to P or T region
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function u=upwardslope(mwi_output)
% This function calculates the start and the end points of upward slopes of the moving window integrator ouyput
% mwi_output=input
% u         = arrays(2,:) of points
% Revised on 1/17/2000 by Srikanth.T.
%-------------------------------------------------------------------------
function [P1,P,P2,Qs,Q,R,S,J,T1,T,T2]=fid_ram_ram(ecg,Fs)
% This function makes use of two ways of calculating the R point locations.
% Inputs are ecg        = input signal and
% Fs = sampling frequency
% By Srikanth.T. on Jan 19, 2000.
%-------------------------------------------------------------------------
function [c1,ind]=localvalley1(dat,thresh)
% This function checks if the given wave is inverted (or) not. The definition of local valley should satisfy and index value is also
% known.
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function [c1,ind]=localvalley1(dat,thresh)
% This function checks if the given wave is inverted (or) not.
% The definition of local valley should satisfy and index value is also known.
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function [c1,ind]=localvalley(dat,thresh)
% This function checks if the given wave is inverted (or) not.
% The definition of local valley should satisfy and index value is also known.
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function c=localpeak(dat,thresh)
% This function checks to see if a particular point is a local peak
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function HR=heartrate(x,y)
% Thisfunction calculates the heartrate based on the SA nodal firing. Heart rate is assumed to be roughly the reciprocal of index of
% present zerocrossing point (ZCP)- last zerocrossing point (ZCP)(RR interval). Scaling factor of 60 and a factor due to sampling
% interval is needed to calculate exactly.
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function y = firstderiv(x,order)
% This function calculates the first derivative of x
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function [y1,y2]=first_diff(ecg)
% This function provides the slope of the ecg signal and applies a threshold and allows only few components.
% ecg = input and
% y1 = difference output
% y2 = thresholded output
% Written by Srikanth.T. on Nov 7, 1999
%-------------------------------------------------------------------------
function [P,Qs,Q,R,S,J,T]=fiducial(ecg,Fs)
% This function estimates the location of P peak, Q wave start, Q
% wave end, R point, S wave start, J point and T peak.
% Inputs are
% ecg = ecg data segment 5 to 10 beats length
```

```
% u   = upwardslope start and end points of mvi output
% Fs  = sampling frequency
% This function tries to detect the presence or absence of individual components namely P. QRS and T waves.  If a wave is not
there.
% corresponding fiducial points remain zero. Rhythm disturbances are viewed as made up of three classes: with absent P waves.
with
% absent T waves and all wave components intact.
% Revised on 3/15/00 by Srikanth.T.
%-------------------------------------------------------------------------
function y=fidplot1(ecg,P,Qs,Q,R,S,J,T)
% This function plots the given data with detected fiducial points, given by fid
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function y=fidplot(ecg,P1,P,P2,Qs,Q,R,S,J,T1,T,T2)
% This function plots the given data with detected fiducial points, given by fid
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function [P,Qs,Q,R,S,J,T]=fidfinal(ecg,Fs)
% This function makes use of two ways of calculating the R point locations.
% Inputs are ecg        = input signal and
% Fs = sampling frequency
% By Srikanth.T. on Nov 16, 1999.
%-------------------------------------------------------------------------
function [mcv,sum]=movcv(x,y)
% This function calculates moving covariance function between two sequences x and y.  x and y may be same or different.  This
% function can be used to estimate the delay of one signal from another similar signal.
% x = sequence 1 and
% y = sequence 2.
% Developed by T. Srikanth 5/9/2000
%-------------------------------------------------------------------------
function c=signchange(x,order)
% length should be even;(ideal 2);
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function c=roughpeakcheck(dat1,m)
% This function checks to see if a point is located on top of a locally elevated region use around 20 points for T peak detection.
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function r=remove(s,array_index,element)
% This function removes the unwanted rows from the matrix of feature values.
% array_index= the particular column to check
% element = Check number (1 or 2 or 3 or 4)
% Designed by Srikanth.T. on April 2, 2000.
%-------------------------------------------------------------------------
function R=r_mwi(ecg,Fs)
% This function estimates the location of R point.
% Inputs are
% ecg = ecg data segment 5 to 10 beats length
% u   = upwardslope start and end points of mvi output
% Fs  = sampling frequency
% Revised on 12/19/99 by Srikanth.T.
%-------------------------------------------------------------------------
function R=r_final(ecg,Fs)
% This function makes use of two ways of calculating the R point locations.
% Inputs are ecg = input signal and
% Fs = sampling frequency
% By Srikanth.T. on Dec 19, 1999.
%-------------------------------------------------------------------------
function R=r_diff(ecg,u2)
% This function calculates the ECG R point.
% Revised by Srikanth.T. on Dec 19, 1999.
%-------------------------------------------------------------------------
function [Pb,Tb]=pt_upslope(mwi_output,start_P,no_R)
% This function calculates the start and the end points of upward slopes of the moving window integrator output
% mwi_output = input
% Pb & Tb               = arrays(2,:) of points
% start_P   = starting P point value
% no_R                  = number of R points detected
% Algorithm starts to look from around 30 points before the start_P mentioned.
% Revised on 12/28/99 by Srikanth.T.
```

```
%-----------------------------------------------------------------------
function [y1,y2,y3]=mwi(x,M)
% x = input ECG (detrended)
% y1 = 5 point differentiated signal
% y2 = squared output
% y3 = moving window integrator output
% M = length of integrator window
% Revised on 9/1/99 by Srikanth.T.
%-----------------------------------------------------------------------
function [y_sq,y_st,y_ab]=slowcurve(dat)
% This function finds if the dat satisfies the conditions for a curve fit  for a region belonging to P or T region.
% Revised on 9/1/99 by Srikanth.T.
%-----------------------------------------------------------------------
function [y1,y2,y3]=mvi(x,M)
% x = input ECG (detrended)
% y1 = 5 point differentiated signal
% y2 = squared output
% y3 = moving window integrator output
% M = length of integrator window
% Revised on 9/1/99 by Srikanth.T.
%-----------------------------------------------------------------------
function y=cleanup(x)
% This function cleans up the matrix u
% Developed by Srikanth.T. on Jan 2 , 2000.
%-----------------------------------------------------------------------
function c=checkvalley(dat)
% This function checks if there is a local peak among the data points.
% Developed by Srikanth.T. on Jan 12' 2000.
%-----------------------------------------------------------------------
function [c,ind]=checksmallpeak(dat,mb)
% This function checks to see if a particular point is a local peak
% Revised on 2/22/00 by Srikanth.T.
%-----------------------------------------------------------------------
function c=checkregion(i,range)
% This function returns 1 if the numerical value of i lies in the range or returns 0.
% Revised on 9/1/99 by Srikanth.T.
%-----------------------------------------------------------------------
function c=checkpeak(dat)
% This function checks if there is a local peak among the data points.
% Developed by Srikanth.T. on Jan 12' 2000.
%-----------------------------------------------------------------------
function m=baselinediff(ecg,R)
% This function calculates the average baseline value of the ecg signal, after the removal of QRS complexes. Inputs are:
% ecg = ecg signal and
% R       = upwardslope start and end points.
% Revised on 1/17/00 by Srikanth.T.
%-----------------------------------------------------------------------
function m=baselineamp(ecg,u)
% This function calculates the average baseline value of the ecg signal, after the removal of QRS complexes. Inputs are:
% ecg = ecg signal and
% u       = upwardslope start and end points.
% Revised on 9/2/99 by Srikanth.T.
%-----------------------------------------------------------------------
function [m1,m2]=baseline(ecg,u)
% This function calculates the mean values of baseline alone neglecting the QRS complexes as also the mean value of the entire
wave.
% Inputs are
% ecg = ecg wave
% u = upwardslope values corresponding to the start and end points of QRS complex.
% By Srikanth.T. on 9/1/99
%-----------------------------------------------------------------------
function code=asccheck(x)
% This function checks if the data segment x is ascending or not. Restrict the check to 3 or 4 points.
% Revised on 9/1/99 by Srikanth.T.
%-----------------------------------------------------------------------
function [c,index]=anglethresh(dat,t1,t2)
% This function calculates the angles between three points and find the value of c. if the angle is above 60 degrees, or pi/3 radians,
% c=1; First step is to calculate the angle.
% Revised on 11/9/99 by Srikanth.T.
%-----------------------------------------------------------------------
```

```
function [theta,a,sa]=angle1(dat)
% This function calculates the angles between three points. This process is continued for first 6 points and a mean and a standard
% deviation is calculated.
% Revised on 11/9/99 by Srikanth.T.
%-------------------------------------------------------------------------
function c=below_above(i,range)
% This function returns 1 if the numerical value of i lies below the range or else returns 0.
% Revised on 11/11/99 by Srikanth.T.
%-------------------------------------------------------------------------
function p_points=definep(data,u,Fs)
% Add an option for two or three arguements, depending on the addition of Fs. if Fs is added, use it to calculate the blank region for
% checking after the occurence of a P like wave. This function defines P wave characteristics and tries to fit in the observed ECG
% segment, data.
% u=an array of size [2*no_of_beats], of QRS start and end points (tentative)
% P wave start point is assumed to be before QRS complex and made up of relatively straight line followed by a smooth curve.
% Curve may be inverted but should broadly satisfy some criteria: A similar inverted portion should follow the initial segment.
% Several P wave morphologies are tested in order to accomadate as much variation as possible.
% Developed by Srikanth.T. on Aug 4, 1999.
%-------------------------------------------------------------------------
function [u3,P,Qs,Q,R,S,J,T]=fid_diff_ram(ecg,u2)
% This function calculates the ECG fiducial points like P wave peaks, QRS complex points and T wave peaks.
% Revised by Srikanth.T. on Jan 17, 2000.
%-------------------------------------------------------------------------
function [P,Qs,Q,R,S,J,T]=fid_final(ecg,Fs)
% This function makes use of two ways of calculating the R point locations.
% Inputs are ecg = input signal and
% Fs = sampling frequency
% Developed by Srikanth.T. on Jan 19, 2000.
%-------------------------------------------------------------------------
function [u3,P,Qs,Q,R,S,J,T]=fid_diff(ecg,u2)
% This function calculates the ECG fiducial points like P wave peaks, QRS complex points and T wave peaks.
% Revised by Srikanth.T. on Mar 21, 2000.
%-------------------------------------------------------------------------
function code=descheck(x)
% This function checks if the data segment x is descending or not. Restrict the check to 5 or 7 or 9 points.
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function code=descendcheck(x,f,f1)
% This function checks if the data segment x is ascending or not. Restrict the check to 5 or 7 or 9 points. f = above 1.00;
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function [c,index]=definitecurve(dat,m)
% This function tests if the given data segment is really a curve. Based on the fact that, a curve is supposed to contain an
% ascendngpart(or) descending part followed by a peak or a valley and again an ascending or descending portion.
% Revised on 9/1/99 by Srikanth.T.
%-------------------------------------------------------------------------
function [P1,P,P2,Qs,Q,R,S,J,T1,T,T2,y]=fid_ram(ecg,Fs)
% This function makes use of two ways of calculating the R point locations.
% Inputs are
% ecg = input signal and
% Fs = sampling frequency
% Developed by Srikanth.T. on Mar 7, 2000.
%-------------------------------------------------------------------------
```

# APPENDIX E

# PERFORMANCE COMPARISION FOR TRAINING VARIOUS STRUCTURES OF FEED-FORWARD NEURAL NETWORKS

# APPENDIX E

# PERFORMANCE COMPARISON FOR TRAINING VARIOUS STRUCTURES OF FEED-FORWARD NEURAL NETWORKS

Smaller data set usually indicates a training set size less than 3-5 times the total number of connections. Final adequate data set contained more than 5-10 times the total number of connections. One epoch represents a complete training step for the data set. Stability is measured in terms of the ability to maintain an achieved error level consistently.

Table. E.1. Performance indicators of ANN's in classifying set1 data – NOR, RBBB, LBBB and APC*

| Name | Structure | Epochs | Goal (mse) | Achieved (mse) | Stability | Comments |
|---|---|---|---|---|---|---|
| Nnset1_1 | [4,3]' | 200 | 0.1 | 0.5445 | Yes | No (Smaller data set) |
| Nnset1_2 | [6,3]' | 200 | 0.1 | 0.3745 | Yes | No (Smaller data set) |
| Nnset1_3 | [8,3]' | 200 | 0.1 | 0.2564 | Yes | No (Smaller data set) |
| Nnset1_4 | [4,3]' | 200 | 0.1 | 0.4996 | Yes | No (Smaller data set) |
| Nnset1_5 | [6,3]' | 200 | 0.1 | 0.4002 | Yes | No (Smaller data set) |
| Nnset1_6 | [8,3]' | 200 | 0.1 | 0.3065 | Yes | No (Smaller data set) |
| Nnset1_7 | [4,3]' | 200 | 0.1 | 0.4315 | Yes | No (Smaller data set) |
| Nnset1_8 | [6,3]' | 200 | 0.1 | 0.4254 | Yes | No (Smaller data set) |
| Nnset1_9 | [8,3]' | 200 | 0.1 | 0.3752 | Yes | No (Smaller data set) |
| Nnset1_10 | [4,3]' | 200 | 0.1 | 0.5730 | Yes | No (Smaller data set) |
| Nnset1_11 | [6,3]' | 200 | 0.1 | 0.3645 | Yes | No (Smaller data set) |
| Nnset1_12 | [8,3]' | 200 | 0.1 | 0.3201 | Yes | No (Smaller data set) |
| Nnset1_13 | [4,3]' | 200 | 0.1 | 0.5068 | Yes | No (Smaller data set) |
| Nnset1_14 | [6,3]' | 200 | 0.1 | 0.3340 | Yes | No (Smaller data set) |
| Nnset1_15 | [8,3]' | 200 | 0.1 | 0.2583 | Yes | No (Smaller data set) |
| Nnset1_16 | [4,3]' | 200 | 0.1 | 0.4842 | Yes | No (Smaller data set) |
| Nnset1_17 | [6,3]' | 200 | 0.1 | 0.2926 | Yes | No (Smaller data set) |
| Nnset1_18 | [8,3]' | 200 | 0.1 | 0.2891 | Yes | No (Smaller data set) |
| Nnset1_19 | [4,3]' | 200 | 0.1 | 0.5288 | Yes | No (Smaller data set) |
| Nnset1_20 | [6,3]' | 200 | 0.1 | 0.4827 | Yes | No (Smaller data set) |
| Nnset1_21 | [8,3]' | 200 | 0.1 | 0.3825 | Yes | No (Smaller data set) |
| Nnset1_22 | [4,3]' | 200 | 0.1 | 0.4984 | Yes | No (Smaller data set) |
| Nnset1_23 | [6,3]' | 200 | 0.1 | 0.4099 | Yes | No (Smaller data set) |
| Nnset1_24 | [8,3]' | 200 | 0.1 | 0.2992 | Yes | No (Smaller data set) |
| Nnset1_1 | [10,3]' | 200 | 0.1 | 0.1409 | Yes | No (Smaller data set) |
| Nnset1_2 | [12,3]' | 200 | 0.1 | 0.1806 | Yes | No (Smaller data set) |
| Nnset1_3 | [10,3]' | 200 | 0.1 | 0.2745 | Yes | No (Smaller data set) |
| Nnset1_4 | [12,3]' | 200 | 0.1 | 0.3304 | Yes | No (Smaller data set) |
| Nnset1_5 | [10,3]' | 200 | 0.1 | 0.2917 | Yes | No (Smaller data set) |

| | | | | | | |
|---|---|---|---|---|---|---|
| Nnset1_6 | [12,3]' | 200 | 0.1 | 0.1987 | Yes | No (Smaller data set) |
| Nnset1_7 | [10,3]' | 200 | 0.1 | 0.2205 | Yes | No (Smaller data set) |
| Nnset1_8 | [12,3]' | 200 | 0.1 | 0.2208 | Yes | No (Smaller data set) |
| Nnset1_9 | [10,3]' | 200 | 0.1 | 0.1972 | Yes | No (Smaller data set) |
| Nnset1_10 | [12,3]' | 200 | 0.1 | 0.1839 | Yes | No (Smaller data set) |
| Nnset1_11 | [10,3]' | 200 | 0.1 | 0.2725 | Yes | No (Smaller data set) |
| Nnset1_12 | [12,3]' | 200 | 0.1 | 0.2304 | Yes | No (Smaller data set) |
| Nnset1_13 | [10,3]' | 200 | 0.1 | 0.2713 | Yes | No (Smaller data set) |
| Nnset1_14 | [12,3]' | 200 | 0.1 | 0.2605 | Yes | No (Smaller data set) |
| Nnset1_15 | [10,3]' | 200 | 0.1 | 0.2300 | Yes | No (Smaller data set) |
| Nnset1_16 | [12,3]' | 200 | 0.1 | 0.2108 | Yes | No (Smaller data set) |
| Nnset1_1 | [8,4] | 500 | 0.1 | 0.3465 | Yes | No (Smaller data set) |
| Nnset1_2 | [10,4] | 500 | 0.1 | 0.2840 | Yes | No (Smaller data set) |
| Nnset1_3 | [8,4] | 500 | 0.1 | 0.2585 | Yes | No (Smaller data set) |
| Nnset1_4 | [10,4] | 500 | 0.1 | 0.1980 | Yes | No (Smaller data set) |
| Nnset1_1 | [13,3]' | 200 | 0.1 | 0.2008 | Yes | No (Smaller data set) |
| Nnset1_2 | [14,3]' | 200 | 0.1 | 0.1736 | Yes | No (Smaller data set) |
| Nnset1_3 | [13,3]' | 200 | 0.1 | 0.2038 | Yes | No (Smaller data set) |
| Nnset1_4 | [14,3]' | 200 | 0.1 | 0.2028 | Yes | No (Smaller data set) |
| Nnset1_5 | [13,3]' | 200 | 0.1 | 0.2342 | Yes | No (Smaller data set) |
| Nnset1_6 | [14,3]' | 200 | 0.1 | 0.1605 | Yes | No (Smaller data set) |
| Nnset1_7 | [13,3]' | 200 | 0.1 | 0.1868 | Yes | No (Smaller data set) |
| Nnset1_8 | [14,3]' | 200 | 0.1 | 0.1858 | Yes | No (Smaller data set) |
| Nnset1_9 | [13,3]' | 200 | 0.1 | 0.1638 | Yes | No (Smaller data set) |
| Nnset1_10 | [14,3]' | 200 | 0.1 | 0.1517 | Yes | No (Smaller data set) |
| Nnset1_11 | [13,3]' | 200 | 0.1 | 0.2233 | Yes | No (Smaller data set) |
| Nnset1_12 | [14,3]' | 200 | 0.1 | 0.1299 | Yes | No (Smaller data set) |
| Nnset1_13 | [13,3]' | 200 | 0.1 | 0.2315 | Yes | No (Smaller data set) |
| Nnset1_14 | [14,3]' | 200 | 0.1 | 0.1914 | Yes | No (Smaller data set) |
| Nnset1_15 | [13,3]' | 200 | 0.1 | 0.1824 | Yes | No (Smaller data set) |
| Nnset1_16 | [14,3]' | 200 | 0.1 | 0.1628 | Yes | No (Smaller data set) |
| Nnset1_1 | [13,4] | 200 | 0.1 | 0.2429 | Yes | No |
| Nnset1_2 | [14,4] | 200 | 0.1 | 0.2361 | Yes | No |
| Nnset1_3 | [13,4] | 200 | 0.1 | 0.2704 | Yes | No |
| Nnset1_4 | [14,4] | 200 | 0.1 | 0.3614 | Yes | No |
| Nnset1_5 | [13,4] | 200 | 0.1 | 0.2755 | Yes | No |
| Nnset1_6 | [14,4] | 200 | 0.1 | 0.2607 | Yes | No |
| Nnset1_7 | [13,4] | 200 | 0.1 | 0.2705 | Yes | No |
| Nnset1_8 | [14,4] | 200 | 0.1 | 0.2673 | Yes | No |
| Nnset1_9 | [13,4] | 200 | 0.1 | 0.2744 | Yes | No |
| Nnset1_10 | [14,4] | 200 | 0.1 | 0.2602 | Yes | No |
| Nnset1_11 | [13,4] | 200 | 0.1 | 0.2528 | Yes | No |
| Nnset1_12 | [14,4] | 200 | 0.1 | 0.2588 | Yes | No |
| Nnset1_13 | [15,4] | 200 | 0.1 | 0.2551 | Yes | No |
| Nnset1_14 | [16,4] | 200 | 0.1 | 0.2246 | Yes | No |
| Nnset1_15 | [16,4] | 200 | 0.1 | 0.2487 | Yes | No |
| Nnset1_16 | [16,4] | 200 | 0.1 | 0.2355 | Yes | No |
| Nnset1_17 | [16,4] | 200 | 0.1 | 0.2573 | Yes | No |
| Nnset1_18 | [16,4] | 200 | 0.1 | 0.2190 | Yes | No |
| Nnset1_141 | [14,4] | 200 | 0.1 | 0.2893 | Yes | No |
| Nnset1_161 | [14,4] | 200 | 0.1 | 0.2809 | Yes | No |
| Nnset1_181 | [14,4] | 200 | 0.1 | 0.2875 | Yes | No |
| Nnset1_18 | [16,4] | 200 | 0.12 | 0.2449 | Yes | No |

| | | | | | | |
|---|---|---|---|---|---|---|
| Nnset1_19 | [16,4] | 200 | 0.12 | 0.2420 | Yes | No |
| Nnset1_20 | [16,4] | 200 | 0.12 | 0.2877 | Yes | No |
| Nnset1_21 | [16,4] | 200 | 0.12 | 0.2822 | Yes | No |
| Nnset1_22 | [16,4] | 200 | 0.12 | 0.2511 | Yes | No |
| Nnset1_23 | [17,4] | 300 | 0.12 | 0.2488 | Yes | No |
| Nnset1_24 | [17,4] | 300 | 0.12 | 0.2444 | Yes | No |
| Nnset1_25 | [17,4] | 200 | 0.15 | 0.2466 | Yes | No |
| Nnset1_26 | [17,4] | 200 | 0.15 | 0.2505 | Yes | No |
| Nnset1_27 | [17,4] | 200 | 0.15 | 0.2181 | Yes | No |
| Nnset1_28 | [17,4] | 200 | 0.15 | 0.2578 | Yes | No |
| Nnset1_29 | [17,4] | 200 | 0.15 | 0.2590 | Yes | No |
| Nnset37_1 | [5,4,4] | 200 | 0.15 | 0.3245 | Yes | No |
| Nnset38_1 | [5,4,4] | 200 | 0.15 | 0.3626 | Yes | No |
| Nnset39_1 | [5,4,4] | 200 | 0.15 | 0.3657 | Yes | No |
| Nnset40_1 | [5,4,4] | 200 | 0.15 | 0.3778 | Yes | No |
| Nnset41_1 | [5,4,4] | 200 | 0.15 | 0.3498 | Yes | No |
| Nnset29_1 | [4,5,4] | 300 | 0.15 | 0.3422 | Yes | No |
| Nnset30_1 | [5,4,4] | 300 | 0.15 | 0.3452 | Yes | No |
| Nnset31_1 | [5,5,4] | 300 | 0.15 | 0.3301 | Yes | No |
| Nnset32_1 | [5,5,4] | 300 | 0.15 | 0.3227 | Yes | No |
| Nnset33_1 | [5,6,4] | 300 | 0.15 | 0.3113 | Yes | No |
| Nnset34_1 | [6,5,4] | 300 | 0.15 | 0.3006 | Yes | No |
| Nnset35_1 | [6,6,4] | 300 | 0.15 | 0.3067 | Yes | No |
| Nnset36_1 | [6,4,4] | 300 | 0.15 | 0.3122 | Yes | No |

[!] APC class not included in this set.
[*] Feature set not mentioned in the Table. They show permutations of the original set of 11 variables.
[*] Only successful neural networks are evaluated with test data.

Table. E.2. Performance indicators of ANN's in classifying **set2** data – PVC,VES, and Paced

| Name | Structure | Epoch | Goal (mse) | Achieved (mse) | Stability |
|---|---|---|---|---|---|
| nnset2_1 | [16,3] | 200 | 0.15 | 0.1153 | Yes |
| nnset2_2 | [16,3] | 200 | 0.15 | 0.1351 | Yes |
| nnset2_3 | [16,3] | 200 | 0.15 | 0.1257 | Yes |
| nnset2_4 | [16, 3] | 200 | 0.15 | 0.13946 | Yes |
| nnset2_5 | [16, 3] | 200 | 0.15 | 0.1350 | Yes |
| nnset2_6 | [16, 3] | 200 | 0.15 | 0.09956 | Yes |
| nnset2_7 | [16, 3] | 200 | 0.15 | 0.1251 | Yes |
| nnset2_8 | [16, 3] | 200 | 0.15; | 0.13966 | Yes |
| nnset2_9 | [16,3] | 200 | 0.15 | 0.1236 | Yes |
| nnset2_10 | [16,3] | 200 | 0.15 | 0.10265 | Yes |
| nnset2_11 | [8,3] | 200 | 0.15 | 0.1280 | Yes |
| nnset2_12 | [8, 3] | 200 | 0.15 | 0.1357 | Yes |
| nnset2_13 | [8, 3] | 200 | 0.15 | 0.1383 | Yes |
| nnset2_14 | [8, 3] | 200 | 0.15 | 0.14263 | Yes |
| nnset2_15 | [8, 3] | 200 | 0.15 | 0.1303 | Yes |
| nnset2_16 | [8, 3] | 200 | 0.15; | 0.12248 | Yes |
| nnset2_17 | [8,3] | 200 | 0.15 | 0.13640 | Yes |
| nnset2_18 | [8,3] | 200 | 0.15 | 0.144134 | Yes |
| nnset2_19 | [8, 3] | 200 | 0.15 | 0.12409 | Yes |
| nnset2_20 | [8, 3] | 200 | 0.15 | 0.14474 | Yes |

| | | | | | |
|---|---|---|---|---|---|
| nnset2_21 | [6, 3] | 200 | 0.15 | 0.14950 | Yes |
| nnset2_22 | [6, 3] | 200 | 0.15 | 0.13950 | Yes |
| nnset2_23 | [6, 3] | 200 | 0.15; | 0.13030 | Yes |
| nnset2_24 | [6, 3] | 200 | 0.15 | 0.1453 | Yes |
| nnset2_25 | [6, 3] | 200 | 0.15 | 0.1381 | Yes |
| nnset2_26 | [6, 3] | 200 | 0.15; | 0.1454 | Yes |
| nnset2_27 | [6, 3] | 200 | 0.15 | 0.1186 | Yes |
| nnset2_28 | [6, 3] | 200 | 0.15 | 0.14473 | Yes |
| nnset2_29 | [6, 3] | 200 | 0.15; | 0.1337 | Yes |
| nnset2_30 | [6, 3] | 200 | 0.15 | 0.1227 | Yes |
| nnset2_31 | [4, 3] | 200 | 0.10 | 0.09608 | Yes |
| nnset2_32 | [4, 3] | 200 | 0.10 | 0.09351 | Yes |
| nnset2_33 | [4, 3] | 200 | 0.10 | 0.1432 | Yes |
| nnset2_34 | [4, 3] | 200 | 0.10 | 0.09658 | Yes |
| nnset2_35 | [4, 3] | 200 | 0.10 | 0.0950 | Yes |
| nnset2_36 | [4, 3] | 200 | 0.10 | 0.09356 | Yes |
| nnset2_37 | [4, 3] | 200 | 0.10 | 0.09913 | Yes |
| nnset2_38 | [4, 3] | 200 | 0.10 | 0.09453 | Yes |
| nnset2_39 | [4, 3] | 200 | 0.10 | 0.09980 | Yes |
| nnset2_40 | [4, 3] | 200 | 0.10 | 0.1069 | Yes |
| nnset2_31 | [3, 3] | 200 | 0.10 | 0.0999 | Yes |
| nnset2_32 | [3, 3] | 200 | 0.10 | 0.16560 | Yes |
| nnset2_33 | [3, 3] | 200 | 0.10 | 0.08017 | Yes |
| nnset2_34 | [3, 3] | 200 | 0.10 | 0.1002 | Yes |
| nnset2_35 | [3, 3] | 200 | 0.10 | 0.15712 | Yes |
| nnset2_36 | [3, 3] | 200 | 0.10 | 0.19574 | Yes |
| nnset2_37 | [3, 3] | 200 | 0.10 | 0.18587 | Yes |
| nnset2_38 | [3, 3] | 200 | 0.10 | 0.11434 | Yes |
| nnset2_39 | [3, 3] | 200 | 0.10 | 0.09342 | Yes |
| nnset2_40 | [3, 3] | 200 | 0.10 | 0.09854 | Yes |
| nnset2_41 | [2, 3] | 200 | 0.10 | 0.20959 | Yes |
| nnset2_42 | [2, 3] | 200 | 0.10 | 0.2312 | Yes |
| nnset2_43 | [2, 3] | 200 | 0.10 | 0.23037 | Yes |
| nnset2_44 | [2, 3] | 200 | 0.10 | 0.1041 | Yes |
| nnset2_45 | [2, 3] | 200 | 0.10 | 0.21439 | Yes |
| nnset2_46 | [3, 3] | 200 | 0.10 | 0.23950 | Yes |
| nnset2_47 | [2, 3] | 200 | 0.10 | 0.20205 | Yes |
| nnset2_48 | [2, 3] | 200 | 0.10 | 0.12860 | Yes |
| nnset2_49 | [2, 3] | 200 | 0.10 | 0.16146 | Yes |
| nnset2_50 | [2, 3] | 200 | 0.10 | 0.13570 | Yes |

Table. E.3. Performance indicators of ANN's in classifying **set3** data – PVC, VES, and Paced

| Name | Structure | Epoch | Goal (mse) | Achieved (mse) | Stability |
|---|---|---|---|---|---|
| Nnset3_1 | [4,3] | 200 | 0.10 | 0.3038 | Yes |
| Nnset3_2 | [4,3] | 200 | 0.10 | 0.3019 | Yes |
| Nnset3_3 | [4,3] | 200 | 0.10 | 0.3468 | Yes |
| Nnset3_4 | [4,3] | 200 | 0.10 | 0.3261 | Yes |
| Nnset3_5 | [4,3] | 200 | 0.10 | 0.3377 | Yes |
| Nnset3_6 | [4,3] | 200 | 0.10 | 0.3408 | Yes |
| Nnset3_7 | [4,3] | 200 | 0.10 | 0.2793 | Yes |
| Nnset3_8 | [4,3] | 200 | 0.10 | 0.2769 | Yes |

| | | | | | |
|---|---|---|---|---|---|
| Nnset3_9 | [4,3] | 200 | 0.10 | 0.3150 | Yes |
| Nnset3_10 | [4,3] | 200 | 0.10 | 0.3373 | Yes |
| Nnset3_11 | [4,3] | 200 | 0.10 | 0.3773 | Yes |
| Nnset3_12 | [4,3] | 200 | 0.10 | 0.3561 | Yes |
| Nnset3_13 | [6,3] | 200 | 0.10 | 0.2498 | Yes |
| Nnset3_14 | [6,3] | 200 | 0.10 | 0.2661 | Yes |
| Nnset3_15 | [6, 3] | 200 | 0.10 | 0.2620 | Yes |
| Nnset3_16 | [6,3] | 200 | 0.10 | 0.3009 | Yes |
| Nnset3_17 | [6,3] | 200 | 0.10 | 0.3092 | Yes |
| Nnset3_18 | [6,3] | 200 | 0.10 | 0.3154 | Yes |
| Nnset3_19 | [6,3] | 200 | 0.10 | 0.2736 | Yes |
| Nnset3_20 | [6,3] | 200 | 0.10 | 0.2704 | Yes |
| Nnset3_21 | [6,3] | 200 | 0.10 | 0.2651 | Yes |
| Nnset3_22 | [6,3] | 200 | 0.10 | 0.3377 | Yes |
| Nnset3_23 | [6,3] | 200 | 0.10 | 0.3024 | Yes |
| Nnset3_24 | [6,3] | 200 | 0.10 | 0.3335 | Yes |
| Nnset3_25 | [8,3] | 200 | 0.10 | 0.2437 | Yes |
| Nnset3_26 | [8,3] | 200 | 0.10 | 0.2544 | Yes |
| Nnset3_27 | [8,3] | 200 | 0.10 | 0.2491 | Yes |
| Nnset3_28 | [8,3] | 200 | 0.10 | 0.2802 | Yes |
| Nnset3_29 | [8,3] | 200 | 0.10 | 0.2895 | Yes |
| Nnset3_30 | [8,3] | 200 | 0.10 | 0.2944 | Yes |
| Nnset3_31 | [8,3] | 200 | 0.10 | 0.2456 | Yes |
| Nnset3_32 | [8,3] | 200 | 0.10 | 0.2760 | Yes |
| Nnset3_33 | [8,3] | 200 | 0.10 | 0.2652 | Yes |
| Nnset3_34 | [8,3] | 200 | 0.10 | 0.2678 | Yes |
| Nnset3_35 | [8,3] | 200 | 0.10 | 0.2873 | Yes |
| Nnset3_36 | [8,3] | 200 | 0.10 | 0.3250 | Yes |

Table. E.4. Performance indicators of ANN's in classifying **set4** data – PVC, VES, and Paced

| Name | Structure | Epoch | Goal (mse) | Achieved (mse) | Stability |
|---|---|---|---|---|---|
| Nnset4_1 | [6,4] | 200 | 0.10 | 0.34821 | Yes |
| Nnset4_2 | [6,4] | 200 | 0.10 | 0.33119 | Yes |
| Nnset4_3 | [6,4] | 200 | 0.10 | 0.3244 | Yes |
| Nnset4_4 | [6,4] | 200 | 0.10 | 0.3409 | Yes |
| Nnset4_5 | [6,4] | 200 | 0.10 | 0.31040 | Yes |
| Nnset4_6 | [6,4] | 200 | 0.10 | 0.3245 | Yes |
| Nnset4_7 | [6,4] | 200 | 0.10 | 0.3119 | Yes |
| Nnset4_8 | [6,4] | 200 | 0.10 | 0.35186 | Yes |
| Nnset4_9 | [6,4] | 200 | 0.10 | 0.28498 | Yes |
| Nnset4_10 | [6,4] | 200 | 0.10 | 0.29909 | Yes |
| Nnset4_11 | [6,4] | 200 | 0.10 | 0.24980 | Yes |
| Nnset4_12 | [6,4] | 200 | 0.10 | 0.275712 | Yes |
| Nnset4_13 | [6,4] | 200 | 0.10 | 0.27250 | Yes |
| Nnset4_14 | [6,4] | 200 | 0.10 | 0.252657 | Yes |
| Nnset4_15 | [8, 4] | 200 | 0.10 | 0.276995 | Yes |
| Nnset4_16 | [8,4] | 200 | 0.10 | 0.25796 | Yes |
| Nnset4_17 | [8,4] | 200 | 0.10 | 0.2713 | Yes |
| Nnset4_18 | [8,4] | 200 | 0.10 | 0.2916 | Yes |
| Nnset4_19 | [8,4] | 200 | 0.10 | 0.2845 | Yes |
| Nnset4_20 | [8,4] | 200 | 0.10 | 0.28728 | Yes |

| | | | | | |
|---|---|---|---|---|---|
| Nnset21_4 | [3,3,3] | 200 | 0.15 | 0.2763 | Yes |
| Nnset22_4 | [4,3,3] | 200 | 0.15 | 0.2528 | Yes |
| Nnset23_4 | [3,4,3] | 200 | 0.15 | 0.2571 | Yes |
| Nnset24_4 | [4,4,3] | 200 | 0.15 | 0.2157 | Yes |
| Nnset25_4 | [2,2,3] | 200 | 0.15 | 0.3085 | Yes |
| Nnset26_4 | [2,3,3] | 200 | 0.15 | 0.2972 | Yes |
| Nnset27_4 | [4,5,3] | 200 | 0.15 | 0.2414 | Yes |
| Nnset28_4 | [5,4,3] | 200 | 0.15 | 0.1761 | Yes |
| Nnset29_4 | [6,1,3] | 200 | 0.15 | 0.4518 | Yes |
| Nnset30_4 | [4,3,4] | 200 | 0.10 | 0.2877 | Yes |
| Nnset31_4 | [4,3,4] | 200 | 0.10 | 0.5356 | Yes |
| Nnset32_4 | [4,3,4] | 200 | 0.10 | 0.324944 | Yes |
| Nnset33_4 | [4,3,4] | 200 | 0.10 | 0.40099 | Yes |
| Nnset34_4 | [4,3,4] | 200 | 0.10 | 0.33727 | Yes |
| Nnset35_4 | [4,3,4] | 200 | 0.10 | 0.34450 | Yes |
| Nnset3_37 | [10,3] | 200 | 0.15 | 0.2303 | Yes |
| Nnset3_38 | [10,3] | 200 | 0.15 | 0.2078 | Yes |
| Nnset3_39 | [10,3] | 200 | 0.15 | 0.2137 | Yes |
| Nnset3_40 | [10,3] | 200 | 0.15 | 0.2177 | Yes |
| Nnset3_41 | [10,3] | 200 | 0.15 | 0.2345 | Yes |
| Nnset3_42 | [10,3] | 200 | 0.15 | 0.2301 | Yes |
| Nnset3_43 | [10,3] | 200 | 0.15 | 0.2163 | Yes |
| Nnset3_44 | [10,3] | 200 | 0.15 | 0.2097 | Yes |
| Nnset3_45 | [10,3] | 200 | 0.15 | 0.1802 | Yes |
| Nnset3_46 | [10,3] | 200 | 0.15 | 0.1930 | Yes |
| Nnset3_47 | [10,3] | 200 | 0.15 | 0.2340 | Yes |
| Nnset3_48 | [10,3] | 200 | 0.15 | 0.2197 | Yes |
| Nnset3_49 | [3,3,3] | 200 | 0.15 | 0.3891 | Yes |
| Nnset3_50 | [4,3,3] | 200 | 0.15 | 0.2593 | Yes |
| Nnset3_51 | [3,4,3] | 200 | 0.15 | 0.2602 | Yes |
| Nnset3_52 | [4,4,3] | 200 | 0.15 | 0.2473 | Yes |
| Nnset3_53 | [2,2,3] | 200 | 0.15 | 0.3143 | Yes |
| Nnset3_54 | [2,3,3] | 200 | 0.15 | 0.3126 | Yes |
| Nnset3_55 | [4,5,3] | 200 | 0.15 | 0.2376 | Yes |
| Nnset3_56 | [5,4,3] | 200 | 0.15 | 0.2284 | Yes |
| Nnset3_57 | [6,1,3] | 200 | 0.15 | 0.2771 | Yes |

# REFERENCES


1. American Heart Association News Letter, http://www.americanheart.org/statistics, 1999.
2. Barro. S., R. Ruiz, and J. Mira, Fuzzy beat labeling for intelligent arrhythmia monitoring, *Computers and Biomedical Research*, Vol. 23, pp. 240-258, 1990.
3. Baxt. W. G., Analysis of clinical variables driving decision in an artificial neural network to identify the presence of myocardial infarction, *Annals in Emergency Medicine*., Vol. 21, pp. 1439-44, Dec, 1992.
4. Baxt. W. G., and J. Skora, Prospective validation of artificial neural network trained to identify acute myocardial infarction, *Lancet*, Vol. 347, pp. 12-15, 1996.
5. Belforte. G., R.D. Mori, and F. Ferraris, A contribution to the automatic processing of electrocardiograms using syntactic methods, *IEEE Transactions on Biomedical Engineering*, Vol. 26, pp. 125-136, March, 1979.
6. Berbari. E. J., D. E. Albert and P. Lander, Spectral estimation of the electrocardiogram, Ed: Coumel. P and O.B. Garfein, 197-208, *The New York Academy of Sciences*, New York, 1990.
7. Bessete. F., and L. Nguyen, Automated electrocardiogram analysis: the state of the art, *Medical Informatics*, Vol. 14, pp. 45-51, 1989.
8. Blackburn. H., History of ECG coding, http://www.epi.umn.edu/Ecg/Pages/history.html ,1999.
9. Blackman. R. B., & J. W. Tukey, The measurement of power spectra, *Dover Publications, Inc*., New York, 1958.
10. Bonadonna. P., Understanding the QT/QTc measurement, Monroe Community College, http://www.monroecc.edu/depts/pstc/parasqt.htm, Rochester, New York, 1998
11. Borovsky. D. and Chr. Zywietz, Feature extraction from a short - term record, ed: H.K. Wolf and F.W. Macfarlane, *Optimization of Computer ECG Processing*, North Holland Publishing Company, IFIP, pp. 109-116, 1980.
12. Casey Klimasauskas. C. C., Neural networks: an engineering perspective, *IEEE Communications Magazine*, pp. 50-53, Sept.,1992.
13. Churchland. P. S., and T. J. Sejnowski, "The computational brain", MIT Press, Cambridge, MA, 1992.
14. Ciaccio. E. J., S. M. Dunn and M. Akay, Bio-signal pattern recognition and interpretation Systems, *IEEE-EMBS Magazine*, pp. 89-97, Sept, 1993.
15. Clayton. R. H., A. Murray and R. W. F. Campbell, Comparison of four techniques for recognition of ventricular fibrillation from the surface ECG, *Medical & Biological Engineering & Computing*, 31, pp. 111-117, 1993.
16. Conover. M. B., Understanding electrocardiography: arrhythmias and the 12-lead ECG, *Mosby Year Book*, 6th Edition, 1992.

17. Cooley. J. W. and J. W. Tukey, An algorithm for the machine computation of complex fourier series, *Mathematics of Computation*, Vol. 19, pp. 297-301, April, 1965.
18. Dallas. J. E., Applied multivariate methods for data analysts, Brookes/Cole Publishing Company, 1998.
19. Degani. R. and G. Bortolan, Methodology of ECG Interpolation in the padova program, *Methods of Information in. Medicine*, 29, pp. 386-392, 1990.
20. Devine. B. and P. W. Macfarlane, Detection of electrocardiographic 'left ventricular strain' using neural nets, *Medical & Biological Engineering & Computing*, 31, pp. 343-348, 1993.
21. Elghazzawi. Z., and F. Geheb, Critique of arrhythmia detectors based on heuristic rules, *Biomedical Instrumentation & Technology*, 31, pp. 263-271, 1997.
22. Fogel. D. B., An information criterion for optimal neural network selection, *IEEE Transactions on Neural Networks*, Vol. 2., pp. 490-497, Sept 1991.
23. Garfein. O. B., The promise of electrocardiography in the 21st century, *Electrocardiography -- Past and Future*, Ed: Coumel. P. and O. B. Garfein, pp. 370-377, New York: The New York Academy of Sciences, 1990.
24. Greenhut. S. E., B. H. Chadi, J. W. Lee, J. M. Jenkins and J. M. Nicklas, An algorithm for the quantification of ST-T segment variability, *Computers and Biomedical Research*, Vol. 22, pp. 339-348, 1989.
25. Hagan. M. T., H.B. Demuth and M. Beale, Neural network design, PWS Publishing Company, 1995.
26. Haisty. W. K., Batchlor. C., Cornfield. J., and Pipberger. H. V., Discriminant function analysis of RR intervals: an algorithm for on-line arrhythmia diagnosis, *Computers and Biomedical Research.*, Vol.5, pp. 247, 1972.
27. Han. S., Classification of cardiac arrhythmias using fuzzy ARTMAP, Ph.D. Dissertation, Department of Electrical Engineering, Florida Institute of Technology, Melbourne, Florida, 1993.
28. Hatala. R., G. R. Norman, and L. R. Brooks, Impact of a clinical scenario on accuracy of electrocardiogram interpretation, *Journal of General Internal Medicine*, Vol. 14, pp. 126-129, 1999.
29. Haykin. S., Neural networks -- a comprehensive foundation, 2nd Edition, Prentice Hall, NJ, 1999.
30. Heden Bo., Analysis of electrocardiograms using artificial neural networks, Doctoral Dissertation, Department of Clinical Physiology, Lund University, Sweden, 1996.
31. Heden Bo., H. Ohlin, R. Rittner, and L. Edenbrandt, Acute myocardial infarction detected in the 12-lead ECG by artificial neural networks, *Circulation*, Vol. 96, pp. 1798-1802, 1997.
32. Heden. Bo, M. Ohlsson, R. Rittner, O. Pahlm, W. K. Haisty, C. Peterson, and L. Edenbrandt, Agreement between artificial neural networks and experienced electrocardiographer on electrocardiographic diagnosis of healed myocardial infarction, *Journal of American College of Cardiology*, Vol. 28, pp. 1012-6, 1996.
33. HME LifePulse Patient Monitor *etc.*, http://www.defib.demon.co.uk/hme4.htm , 1999

34. Hornik. K., Approximation capabilities of multilayer feedforward networks, *Neural Networks*, Vol.4., pp. 251-257, 1991.
35. Jain. V., and P.M. Rautaharju, The use of discretized features for classification of Electrocardiograms, *Optimization of Computer ECG Processing*, ed: H.K. Wolf and F.W. Macfarlane, North Holland Publishing Company, IFIP, pp. 117-120, 1980.
36. Jenkins. D., ECG database, http://homepages.enterprise.net/djenkins/ecghome.htm#contents, 1998
37. Katz. A. M., Physiology of the heart, Second Edition, New York: Raven Press, 1992.
38. Kay. S. M., Modern spectral estimation, Prentice-Hall Signal Processing Series, NJ, 1992.
39. Kelly. S. J., ECG interpretation: identifying arrhythmia, Philadelphia: J. B. Lippincott Co., 1984.
40. Keselbrener. L., M. Keselbrener and S. Akselrod, Nonlinear high pass filter for R-wave detection in ECG signal, *Medical Engineering &Physics*, vol. 19, pp. 481-484, 1997.
41. Klingeman. J. and H.V. Pipberger, Computer classification of electrocardiograms, *Computers and Biomedical Research*, Vol. 1, pp. 1-17, 1967.
42. Kors. JA, G. van Herpen, J. Wu, Z. Zhang, R. J. Prineas, J. H. van Bemmel, Validation of a new computer program for Minnesota coding, *Journal of Electrocardiology*, Vol. 29, Supplement, pp. 83-88, 1996.
43. Kosko. B., Neural networks and fuzzy Systems, 1st Edition, Prentice Hall, NJ, 1992.
44. Kuppuraj. R. N., An hybrid AI approach to arrhythmia diagnosis, Ph.D. Dissertation, Louisiana Tech University, Ruston, LA, 1995.
45. Laguna. P., R. Jane, and P. Caminal, Automatic detection of wave boundaries in multilead ECG signals: Validation with CSE database, *Computers and Biomedical Research*, Vol. 27, pp. 45-60, 1994.
46. Laguna. P., R.G. Mark, A. Goldberg, and G. B. Moody, A database for evaluation of algorithms for measurement of QT and other waveform intervals in the ECG, *Proceedings of Computers in Cardiology*, Vol. 24, pp. 673-676, 1997.
47. Lancaster and Salkauskas, Curve and surface fitting: an introduction, 1986.
48. Langberg. J. L., J.C. Burtle, and K.K. McTeague, Spectral Analysis of Electrocardiogram Predicts Recurrence of Atrial Fibrillation after Cardioversion, *Journal of Electrocardiology*, Vol. 31, Supplement, pp. 80-84, 1998.
49. Leblanc. A. R., Quantitative Analysis of Cardiac Arrhythmia, *CRC Critical Reviews in Biomedical Engineering*, Vol. 14, pp. 1-60, 1989.
50. Lee. S. C., Using translation-invariant neural network to diagnose heart arrhythmia, *Proceedings of 11th Annual International Conference on IEEE-EMBS*, pp. 2025-2026, 1989.
51. Levary. R. R., Rule-based artificial neural networks, *International Journal of Systems Science*, Vol.24, pp. 1415-1418, 1993.
52. Li.C., C. Zheng, and C. Tai, Detection of ECG characteristic points using wavelet transforms, *IEEE Transactions on Biomedical Engineering*, Vol. 42, pp. 21-28, 1995.
53. Macfarlane. P. W. and T.D.V. Lawrie, An introduction to automated electrocardiogram interpretation, Butterworths & Co. (Publishers) Ltd, 1974.

54. Maren. A., C. Harston, and R. Pap -- Handbook of neural computing applications -- Academic Press, Inc., 1990.
55. Marple. L., Digital spectral analysis, Prentice Hall, NJ, 1987.
56. Marquette ECG Machines, http://www.mei.com/products/area.all.html, 1999.
57. McLaughlin. N. B., R.W.F. Campbell and A. Murray, Accuracy of Automatic QT Measurement Techniques, *Proceedings of Computers in Cardiology*, pp.863-866, Sept 1993.
58. Medtronic 9790 Programmer, Implantable Cardioverter-Defibrillator, Medtronic, Inc., http://www.medtronic.com/tachy/clinician/programmer/9790prog.html , 1999.
59. Minami. K., H. Nakajima, and T. Toyashima, Real-time discrimination of ventricular tachyarrhythmia with fourier-transform neural network, *IEEE Transactions on Biomedical Engineering*, Vol. 46, pp. 179-185, 1999.
60. Moody. G. B., MIT-BIH Arrhythmia Database Directory, Harvard University-MIT Division of Health Sciences and Technology, 1992.
61. Murata. N., S. Yoshizawa and S. Amari, Network Information Criterion -- Determining the number of hidden units for an Artificial Neural Network Model, *IEEE Transactions on Neural Networks*, Vol. 5, Nov, 1994.
62. Murthy. I. S. N. and U. C. Niranjan, Component wave delineation of ECG by filtering in the Fourier domain, *Medical & Biological Engineering & Computing*, Vol. 30, pp. 169-175, 1992.
63. Ohlsson. M., Combinatorial Optimization and Artificial Neural Networks, Doctoral Dissertation, Department of Theoretical Physics, Lund University, Sweden, 1995.
64. Okajima. O.,N. Okamoto, M. Yokoi, T. Iwatsuka and N. Ohsawa, Methodology of ECG Interpretation in the Nagoya Program, *Methods of Information in Medicine*, Vol. 29, pp. 341-345, 1990.
65. Pahlm. O. and L. Sornmo, Software QRS detection in ambulatory monitoring -- a review, *Medical & Biological Engineering & Computing*, Vol. 22, pp. 289- 297, 1984.
66. Pan. J. and W. Tompkins, A Real Time QRS Detection Algorithm, *IEEE Transactions on Biomedical Engineering*, Vol. 32, pp.230-235,1985.
67. Pipberger. H. V., Computer analysis of the electrocardiogram, *In Computers and Biomedical Research*, Ed: B. Waxman and R. Stagy, pp. 377, New York: Academic Press, 1965.
68. Pipberger. H. V., D. McCaughan, D. Littman, H.A. Pipberger, J. Cornfield, R.A. Dunn, C.D. Batchlor, and A. S. Berson, Clinical application of a second generation of electrocardiographic computer program, *American Journal of Cardiology*, Vol. 35, pp. 597-608, 1975.
69. Pipberger. H. V., R. J. Arms and F. W. Stallman, Automatic screening of normal and abnormal electrocardiograms by means of a digital computer. *Proceedings of .Society for .Experts in Biology and Medicine.*, Vol. 106,pp. 130, 1961.
70. Rasiah. A. I., R. Togneri, and Y. Attikiouzel, QRS detection using morphological and rhythm information, *IEEE International Conference on Neural Networks*, 1995.
71. Rautaharju. P. M., H. P. Calhoun, and B. R. Chaitman, NOVACODE Serial ECG classification system for clinical trials and epidemiologic studies, *Journal of Electrocardiology*, Vol. 24, Supplement, pp. 179-187, 1992.

72. Rautaharju. P. M., L.P.Park, B.R. Chaitman, F. Rautaharju and Z. Zhang, The novacode criteria for classification of ECG abnormalities and their clinically significant progression and regression, *Journal of Electrocardiology*, Vol. 31, pp. 157-187, 1998.
73. Robinson. E. A., A historical perspective of spectrum estimation, *Proceedings of the IEEE*, Vol. 70, No.9, pp. 885-907, 1982.
74. Saxena. S. C., A. Sharma and S.C. Chaudhary, Data compression and feature extraction of ECG signals, *International Journal of Systems Science*, Vol. 28, pp. 483-498, 1997.
75. Schaffer. R. W. and L.R. Rabiner, A digital processing approach to interpolation, *Proceedings of the IEEE*, Vol. 61, June, 1972.
76. Schalkoff. R.-- Pattern recognition--statistical, structural and neural approaches, John Wiley & Sons, *Inc.*, New York, 1992.
77. Schijivenaars. R. J. A., J.A. Kors, G. van Harpen, J. H. van Bemmel, Assessment of the stability of automated ECG interpretation, Annual Report, http://www.eur.nl/fgg/mi/annrep96/ , 1996.
78. Schimminger. T., Analysis and modeling of ECG signals using nonlinear methods, PhD Dissertation, Technische Universitat Dresden, Denmark, http://circhp.epfl/ch/~schimmin/uni/beleg/bodv.html , 1998.
79. Schuster. A., On the investigation of hidden periodicities with application to a supposed 26 day period of meteriological phenomena, *Terrestrial Magnetism*, Vol. 3., pp. 13-41, March, 1898.
80. Sheffield. L. T., The role of computerized ECG interpretation in clinical trials, *Journal of Electrocardiology - Supplemental Issue*, pp.37-44, 1987.
81. Shkurovich. S., A. V. Sahakian, and S. Swiryn, Detection of atrial activity from high-voltage leads of implantable ventricular fibrillators, using a cancellation technique, *IEEE Transactions on Biomedical Engineering*, Vol. 45, pp. 229-234, 1998.
82. Silipo. R., M. Gori, A. Taddei, M. Varanini and C. Marchesi, Classification of arrhythmic events in ambulatory electrocardiogram using artificial neural networks, *Computers and Biomedical Research*, Vol. 28, pp. 305-318, 1995.
83. Skordalakis. E., Recognition of the shape of the ST segment in ECG waveforms, *IEEE Transactions on Biomedical Engineering*, Vol. 33, pp. 972-974, 1986.
84. Sornmo. L., O. Pahlm, P. O. Borjesson and M. E. Nygards, A method for evaluation of QRS shape features using a mathematical model for the ECG, *IEEE Transactions on Biomedical Engineering*, Vol. 27, pp. 713-717, July, 1980.
85. Srikanth. T., Analysis of HRV Signals in Time-Frequency Domain to Study Cardiac Neuropathy and Meditation Subjects, MS Thesis, Indian Institute of Technology, Chennai, 1996.
86. Srikanth. T., S. A. Napper and H. Gu, A combination approach to frequency domain analysis of electrocardiogram signals, *Proceedings of IEEE-EMBS/*BMES *Joint Meeting*, Atlanta, 1999.
87. Srikanth. T., S. A. Napper and H. Gu, Assessment of re-sampling techniques for electrocardiogram signals for feature extraction, statistical and neural network applications, *Proceedings of Computers in Cardiology*, Cleveland, 1998.

88. Steinbegler. P., R. Haberl, G. Jilge and G. Steinbeck, Single beat analysis of ventricular late potentials from the surface electrocardiogram using the spectrotemporal pattern recognition algorithm in patients with coronary artery disease, *European Heart Journal*, Vol. 19, pp. 435-446, 1989.
89. Steinberg. C. A., S. Abraham and C.A. Caceres, Pattern recognition in the clinical electrocardiogram, *IEEE Transactions on Biomedical Electronics*, Vol. 9, pp. 23-30, 1962.
90. Strohmenger. H. V., K. H. Lindher, and C. G. Brown, Analysis of the ventricular fibrilation ECG signal amplitude and frequency parameters as predictors of countershock success in humans, *Chest*, Vol. 111, pp. 584-589, 1997.
91. Suzuki. Y. and K. Ono, Personal computer system for ECG ST-segment recognition based on neural networks, *Medical & Biological Engineering & Computing*, Vol. 30, pp. 2-8, 1992.
92. Svirbely. J. R., and M. G. Sriram, The Medical Algorithms Project, http://www.medal.org/ , March, 2000.
93. Thakor. N. V., J. G. Webster, and W. J. Tompkins, Estimation of QRS Complex Power Spectra for Design of a QRS Filter, *IEEE Transactions on Biomedical Engineering*, Vol. 31., Nov, 1984.
94. Tong. D. A., A Model-based Interpretation of the Electrocardiogram, Ph.D. Dissertation, Louisiana Tech University, Ruston, LA, 1992.
95. Trahanias. P., and E. Skordalakis, Bottom-up approach to the ECG pattern-recognition problem, *Medical & Biological Engineering & Computing*, Vol. 27, pp. 221-229, 1989.
96. van Bemmel. J. H., Chr. Zywietz and J. A. Kors, Signal analysis for ECG interpretation, *Methods of Information in Medicine*, Vol. 29, pp. 317-329, 1990.
97. Voss. A., J. Kurths and H. Fiehring, Frequency domain analysis of highly amplified ECG on the basis of maximum entropy spectral estimation, *Medical & Biological Engineering & Computing*, Vol. 30, pp. 277-282, 1992.
98. Wagner. G. S., Marriot's practical electrocardiography, Ninth Edition, Williams & Wilkins, 1994.
99. Wasserman. P. D., -- Neural computing--theory and practice, New York: van Nostrand Reinhold, 1989.
100. Werbos. P. J., Beyond regression: new tools for prediction and analysis in the behavioral sciences, PhD Thesis, Harvard University, Cambridge, MA, 1974.
101. Widrow. B. and M. E.Lehr, 30 years of adaptive neural networks: perceptron, madaline, and backpropagation, *Proceedings of the IEEE*, Vol. 78, pp. 1415-1442, 1990.
102. Willems. J. J., C. Abreu-Lima, P. Arnaud, C. R. Brohet, B. Denis, J. Gehring, I. Graham, G. van Harpen, H. Machoda, J. Michelis, and S. Moulopolos, Evaluation of ECG interpretation results obtained by computer and cardiologists, *Methods of Information in Medicine*, Vol. 29, pp. 308-316, 1990.
103. Willems. J. J., Quantitative electrocardiography -- standardization and performance evaluation, *Electrocardiography -- Past and Future*, Ed: Coumel. P. and O. B. Garfein, 329-342, New York: The New York Academy of Sciences, 1990.

104. Willems. J. L., P. Arnaud, J. H. van Bemmel, R. Degani, P. W. Macfarlane, and Chr. Zywietz, Common standards for quantitative electrocardiography : goals and main results, *Methods of Information in Medicine*, Vol. 29, pp. 263-271, 1990.

105. Xue. Q., Y. H. Hu and W. J. Tompkins, Neural-network-based adaptive matched filtering for QRS detection, *IEEE Transactions on Biomedical Engineering*, Vol. 39, pp. 317-329, 1992.

106. Yanowitz, F., P. Kimas, D. Rawling, and H. A. Fozzard, Accuracy of continuous real-time ECG dysrhythmia monitoring system, *Circulation*, Vol. 50, pp. 65-72, 1974.

107. Zahedi. F., An introduction to neural networks and a comparison with artificial intelligence and expert systems, *Interfaces*, Vol. 21, pp. 25-38, April, 1991.

# VITA

Srikanth Thiagarajan was born and brought up in Tamil Nadu state of India. His parents are Mrs. T. Prema and Mr. T.S. Thiagarajan. He has an elder brother and an elder sister. He graduated from Madurai Kamaraj University, India, in July, 1993 with Bachelor of Engineering degree in Electronics and Communication Engineering. He finished Master of Science degree in Biomedical Engineering Division, Indian Institute of Technology, Chennai, India in 1996. He joined Department of Biomedical Engineering, Louisiana Tech University, Ruston, LA as a doctoral student in August, 1996. He became a candidate for doctoral degree by Winter, 1998-99. His future plans after doctoral studies include academic and industrial research related to bio-signal processing, interpretation and organization.